



Junos[®] OS

Layer 2 Configuration Guide

Release
11.2



Published: 2011-05-16

Juniper Networks, Inc.
1194 North Mathilda Avenue
Sunnyvale, California 94089
USA
408-745-2000
www.juniper.net

This product includes the Envoy SNMP Engine, developed by Epilogue Technology, an Integrated Systems Company. Copyright © 1986-1997, Epilogue Technology Corporation. All rights reserved. This program and its documentation were developed at private expense, and no part of them is in the public domain.

This product includes memory allocation software developed by Mark Moraes, copyright © 1988, 1989, 1993, University of Toronto.

This product includes FreeBSD software developed by the University of California, Berkeley, and its contributors. All of the documentation and software included in the 4.4BSD and 4.4BSD-Lite Releases is copyrighted by the Regents of the University of California. Copyright © 1979, 1980, 1983, 1986, 1988, 1989, 1991, 1992, 1993, 1994. The Regents of the University of California. All rights reserved.

GateD software copyright © 1995, the Regents of the University. All rights reserved. Gate Daemon was originated and developed through release 3.0 by Cornell University and its collaborators. Gated is based on Kirton's EGP, UC Berkeley's routing daemon (routed), and DCN's HELLO routing protocol. Development of Gated has been supported in part by the National Science Foundation. Portions of the GateD software copyright © 1988, Regents of the University of California. All rights reserved. Portions of the GateD software copyright © 1991, D. L. S. Associates.

This product includes software developed by Maker Communications, Inc., copyright © 1996, 1997, Maker Communications, Inc.

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.

Products made or sold by Juniper Networks or components thereof might be covered by one or more of the following patents that are owned by or licensed to Juniper Networks: U.S. Patent Nos. 5,473,599, 5,905,725, 5,909,440, 6,192,051, 6,333,650, 6,359,479, 6,406,312, 6,429,706, 6,459,579, 6,493,347, 6,538,518, 6,538,899, 6,552,918, 6,567,902, 6,578,186, and 6,590,785.

Junos® OS Layer 2 Configuration Guide
Release 11.2
Copyright © 2011, Juniper Networks, Inc.
All rights reserved.

Revision History
June 2011—R1 Junos OS 11.2

The information in this document is current as of the date listed in the revision history.

YEAR 2000 NOTICE

Juniper Networks hardware and software products are Year 2000 compliant. The 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

READ THIS END USER LICENSE AGREEMENT ("AGREEMENT") BEFORE DOWNLOADING, INSTALLING, OR USING THE SOFTWARE. BY DOWNLOADING, INSTALLING, OR USING THE SOFTWARE OR OTHERWISE EXPRESSING YOUR AGREEMENT TO THE TERMS CONTAINED HEREIN, YOU (AS CUSTOMER OR IF YOU ARE NOT THE CUSTOMER, AS A REPRESENTATIVE/AGENT AUTHORIZED TO BIND THE CUSTOMER) CONSENT TO BE BOUND BY THIS AGREEMENT. IF YOU DO NOT OR CANNOT AGREE TO THE TERMS CONTAINED HEREIN, THEN (A) DO NOT DOWNLOAD, INSTALL, OR USE THE SOFTWARE, AND (B) YOU MAY CONTACT JUNIPER NETWORKS REGARDING LICENSE TERMS.

1. **The Parties.** The parties to this Agreement are (i) Juniper Networks, Inc. (if the Customer's principal office is located in the Americas) or Juniper Networks (Cayman) Limited (if the Customer's principal office is located outside the Americas) (such applicable entity being referred to herein as "Juniper"), and (ii) the person or organization that originally purchased from Juniper or an authorized Juniper reseller the applicable license(s) for use of the Software ("Customer") (collectively, the "Parties").

2. **The Software.** In this Agreement, "Software" means the program modules and features of the Juniper or Juniper-supplied software, for which Customer has paid the applicable license or support fees to Juniper or an authorized Juniper reseller, or which was embedded by Juniper in equipment which Customer purchased from Juniper or an authorized Juniper reseller. "Software" also includes updates, upgrades and new releases of such software. "Embedded Software" means Software which Juniper has embedded in or loaded onto the Juniper equipment and any updates, upgrades, additions or replacements which are subsequently embedded in or loaded onto the equipment.

3. **License Grant.** Subject to payment of the applicable fees and the limitations and restrictions set forth herein, Juniper grants to Customer a non-exclusive and non-transferable license, without right to sublicense, to use the Software, in executable form only, subject to the following use restrictions:

- a. Customer shall use Embedded Software solely as embedded in, and for execution on, Juniper equipment originally purchased by Customer from Juniper or an authorized Juniper reseller.
- b. Customer shall use the Software on a single hardware chassis having a single processing unit, or as many chassis or processing units for which Customer has paid the applicable license fees; provided, however, with respect to the Steel-Belted Radius or Odyssey Access Client software only, Customer shall use such Software on a single computer containing a single physical random access memory space and containing any number of processors. Use of the Steel-Belted Radius or IMS AAA software on multiple computers or virtual machines (e.g., Solaris zones) requires multiple licenses, regardless of whether such computers or virtualizations are physically contained on a single chassis.
- c. Product purchase documents, paper or electronic user documentation, and/or the particular licenses purchased by Customer may specify limits to Customer's use of the Software. Such limits may restrict use to a maximum number of seats, registered endpoints, concurrent users, sessions, calls, connections, subscribers, clusters, nodes, realms, devices, links, ports or transactions, or require the purchase of separate licenses to use particular features, functionalities, services, applications, operations, or capabilities, or provide throughput, performance, configuration, bandwidth, interface, processing, temporal, or geographical limits. In addition, such limits may restrict the use of the Software to managing certain kinds of networks or require the Software to be used only in conjunction with other specific Software. Customer's use of the Software shall be subject to all such limitations and purchase of all applicable licenses.
- d. For any trial copy of the Software, Customer's right to use the Software expires 30 days after download, installation or use of the Software. Customer may operate the Software after the 30-day trial period only if Customer pays for a license to do so. Customer may not extend or create an additional trial period by re-installing the Software after the 30-day trial period.
- e. The Global Enterprise Edition of the Steel-Belted Radius software may be used by Customer only to manage access to Customer's enterprise network. Specifically, service provider customers are expressly prohibited from using the Global Enterprise Edition of the Steel-Belted Radius software to support any commercial network access services.

The foregoing license is not transferable or assignable by Customer. No license is granted herein to any user who did not originally purchase the applicable license(s) for the Software from Juniper or an authorized Juniper reseller.

4. **Use Prohibitions.** Notwithstanding the foregoing, the license provided herein does not permit the Customer to, and Customer agrees not to and shall not: (a) modify, unbundle, reverse engineer, or create derivative works based on the Software; (b) make unauthorized copies of the Software (except as necessary for backup purposes); (c) rent, sell, transfer, or grant any rights in and to any copy of the Software, in any form, to any third party; (d) remove any proprietary notices, labels, or marks on or in any copy of the Software or any product in which the Software is embedded; (e) distribute any copy of the Software to any third party, including as may be embedded in Juniper equipment sold in the secondhand market; (f) use any 'locked' or key-restricted feature, function, service, application, operation, or capability without first purchasing the applicable license(s) and obtaining a valid key from Juniper, even if such feature, function, service, application, operation, or capability is enabled without a key; (g) distribute any key for the Software provided by Juniper to any third party; (h) use the

Software in any manner that extends or is broader than the uses purchased by Customer from Juniper or an authorized Juniper reseller; (i) use Embedded Software on non-Juniper equipment; (j) use Embedded Software (or make it available for use) on Juniper equipment that the Customer did not originally purchase from Juniper or an authorized Juniper reseller; (k) disclose the results of testing or benchmarking of the Software to any third party without the prior written consent of Juniper; or (l) use the Software in any manner other than as expressly provided herein.

5. **Audit.** Customer shall maintain accurate records as necessary to verify compliance with this Agreement. Upon request by Juniper, Customer shall furnish such records to Juniper and certify its compliance with this Agreement.

6. **Confidentiality.** The Parties agree that aspects of the Software and associated documentation are the confidential property of Juniper. As such, Customer shall exercise all reasonable commercial efforts to maintain the Software and associated documentation in confidence, which at a minimum includes restricting access to the Software to Customer employees and contractors having a need to use the Software for Customer's internal business purposes.

7. **Ownership.** Juniper and Juniper's licensors, respectively, retain ownership of all right, title, and interest (including copyright) in and to the Software, associated documentation, and all copies of the Software. Nothing in this Agreement constitutes a transfer or conveyance of any right, title, or interest in the Software or associated documentation, or a sale of the Software, associated documentation, or copies of the Software.

8. **Warranty, Limitation of Liability, Disclaimer of Warranty.** The warranty applicable to the Software shall be as set forth in the warranty statement that accompanies the Software (the "Warranty Statement"). Nothing in this Agreement shall give rise to any obligation to support the Software. Support services may be purchased separately. Any such support shall be governed by a separate, written support services agreement. TO THE MAXIMUM EXTENT PERMITTED BY LAW, JUNIPER SHALL NOT BE LIABLE FOR ANY LOST PROFITS, LOSS OF DATA, OR COSTS OR PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES, OR FOR ANY SPECIAL, INDIRECT, OR CONSEQUENTIAL DAMAGES ARISING OUT OF THIS AGREEMENT, THE SOFTWARE, OR ANY JUNIPER OR JUNIPER-SUPPLIED SOFTWARE. IN NO EVENT SHALL JUNIPER BE LIABLE FOR DAMAGES ARISING FROM UNAUTHORIZED OR IMPROPER USE OF ANY JUNIPER OR JUNIPER-SUPPLIED SOFTWARE. EXCEPT AS EXPRESSLY PROVIDED IN THE WARRANTY STATEMENT TO THE EXTENT PERMITTED BY LAW, JUNIPER DISCLAIMS ANY AND ALL WARRANTIES IN AND TO THE SOFTWARE (WHETHER EXPRESS, IMPLIED, STATUTORY, OR OTHERWISE), INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NONINFRINGEMENT. IN NO EVENT DOES JUNIPER WARRANT THAT THE SOFTWARE, OR ANY EQUIPMENT OR NETWORK RUNNING THE SOFTWARE, WILL OPERATE WITHOUT ERROR OR INTERRUPTION, OR WILL BE FREE OF VULNERABILITY TO INTRUSION OR ATTACK. In no event shall Juniper's or its suppliers' or licensors' liability to Customer, whether in contract, tort (including negligence), breach of warranty, or otherwise, exceed the price paid by Customer for the Software that gave rise to the claim, or if the Software is embedded in another Juniper product, the price paid by Customer for such other product. Customer acknowledges and agrees that Juniper has set its prices and entered into this Agreement in reliance upon the disclaimers of warranty and the limitations of liability set forth herein, that the same reflect an allocation of risk between the Parties (including the risk that a contract remedy may fail of its essential purpose and cause consequential loss), and that the same form an essential basis of the bargain between the Parties.

9. **Termination.** Any breach of this Agreement or failure by Customer to pay any applicable fees due shall result in automatic termination of the license granted herein. Upon such termination, Customer shall destroy or return to Juniper all copies of the Software and related documentation in Customer's possession or control.

10. **Taxes.** All license fees payable under this agreement are exclusive of tax. Customer shall be responsible for paying Taxes arising from the purchase of the license, or importation or use of the Software. If applicable, valid exemption documentation for each taxing jurisdiction shall be provided to Juniper prior to invoicing, and Customer shall promptly notify Juniper if their exemption is revoked or modified. All payments made by Customer shall be net of any applicable withholding tax. Customer will provide reasonable assistance to Juniper in connection with such withholding taxes by promptly: providing Juniper with valid tax receipts and other required documentation showing Customer's payment of any withholding taxes; completing appropriate applications that would reduce the amount of withholding tax to be paid; and notifying and assisting Juniper in any audit or tax proceeding related to transactions hereunder. Customer shall comply with all applicable tax laws and regulations, and Customer will promptly pay or reimburse Juniper for all costs and damages related to any liability incurred by Juniper as a result of Customer's non-compliance or delay with its responsibilities herein. Customer's obligations under this Section shall survive termination or expiration of this Agreement.

11. **Export.** Customer agrees to comply with all applicable export laws and restrictions and regulations of any United States and any applicable foreign agency or authority, and not to export or re-export the Software or any direct product thereof in violation of any such restrictions, laws or regulations, or without all necessary approvals. Customer shall be liable for any such violations. The version of the Software supplied to Customer may contain encryption or other capabilities restricting Customer's ability to export the Software without an export license.

12. **Commercial Computer Software.** The Software is "commercial computer software" and is provided with restricted rights. Use, duplication, or disclosure by the United States government is subject to restrictions set forth in this Agreement and as provided in DFARS 227.7201 through 227.7202-4, FAR 12.212, FAR 27.405(b)(2), FAR 52.227-19, or FAR 52.227-14 (ALT III) as applicable.

13. **Interface Information.** To the extent required by applicable law, and at Customer's written request, Juniper shall provide Customer with the interface information needed to achieve interoperability between the Software and another independently created program, on payment of applicable fee, if any. Customer shall observe strict obligations of confidentiality with respect to such information and shall use such information in compliance with any applicable terms and conditions upon which Juniper makes such information available.

14. **Third Party Software.** Any licensor of Juniper whose software is embedded in the Software and any supplier of Juniper whose products or technology are embedded in (or services are accessed by) the Software shall be a third party beneficiary with respect to this Agreement, and such licensor or vendor shall have the right to enforce this Agreement in its own name as if it were Juniper. In addition, certain third party software may be provided with the Software and is subject to the accompanying license(s), if any, of its respective owner(s). To the extent portions of the Software are distributed under and subject to open source licenses obligating Juniper to make the source code for such portions publicly available (such as the GNU General Public License ("GPL") or the GNU Library General Public License ("LGPL")), Juniper will make such source code portions (including Juniper modifications, as appropriate) available upon request for a period of up to three years from the date of distribution. Such request can be made in writing to Juniper Networks, Inc., 1194 N. Mathilda Ave., Sunnyvale, CA 94089, ATTN: General Counsel. You may obtain a copy of the GPL at <http://www.gnu.org/licenses/gpl.html>, and a copy of the LGPL at <http://www.gnu.org/licenses/lgpl.html>.

15. **Miscellaneous.** This Agreement shall be governed by the laws of the State of California without reference to its conflicts of laws principles. The provisions of the U.N. Convention for the International Sale of Goods shall not apply to this Agreement. For any disputes arising under this Agreement, the Parties hereby consent to the personal and exclusive jurisdiction of, and venue in, the state and federal courts within Santa Clara County, California. This Agreement constitutes the entire and sole agreement between Juniper and the Customer with respect to the Software, and supersedes all prior and contemporaneous agreements relating to the Software, whether oral or written (including any inconsistent terms contained in a purchase order), except that the terms of a separate written agreement executed by an authorized Juniper representative and Customer shall govern to the extent such terms are inconsistent or conflict with terms contained herein. No modification to this Agreement nor any waiver of any rights hereunder shall be effective unless expressly assented to in writing by the party to be charged. If any portion of this Agreement is held invalid, the Parties agree that such invalidity shall not affect the validity of the remainder of this Agreement. This Agreement and associated documentation has been written in the English language, and the Parties agree that the English version will govern. (For Canada: Les parties aux présentes confirment leur volonté que cette convention de même que tous les documents y compris tout avis qui s'y rattache, soient rédigés en langue anglaise. (Translation: The parties confirm that this Agreement and all related documentation is and will be in the English language)).

Abbreviated Table of Contents

	About This Guide	xxi
Part 1	Overview	
Chapter 1	Layer 2 Services for MX Series Routers Overview	3
Part 2	Configuration Basics for Layer 2 Services	
Chapter 2	Configuring Routing Instances for Layer 2 Services	13
Chapter 3	Configuring Basic Layer 2 Services	21
Chapter 4	Configuring Link Layer Discovery Protocol	31
Chapter 5	Summary of Link Layer Discovery Protocol Configuration Statements ...	37
Chapter 6	Configuring Multiple VLAN Registration Protocol	47
Chapter 7	Summary of Multiple VLAN Registration Protocol Configuration Statements	71
Part 3	Layer 2 Port Mirroring	
Chapter 8	Layer 2 Port Mirroring Overview	85
Chapter 9	Configuring Layer 2 Port Mirroring	101
Chapter 10	Examples of Layer 2 Port Mirroring	127
Part 4	Layer 2 Bridging	
Chapter 11	Layer 2 Bridging Overview	147
Chapter 12	Configuring Layer 2 Bridging	151
Chapter 13	Summary of Layer 2 Bridging Configuration Statements	175
Part 5	Layer 2 Address Learning and Forwarding	
Chapter 14	Layer 2 Learning and Forwarding	195
Chapter 15	Summary of Layer 2 Address Learning and Forwarding Configuration Statements	199
Part 6	Spanning-Tree Protocols	
Chapter 16	Spanning-Tree Protocols Overview	205
Chapter 17	Guidelines for Configuring Spanning-Tree Protocols	209
Chapter 18	Configuring Spanning-Tree Protocols	217
Chapter 19	Spanning-Tree Protocol Options	233

Chapter 20	Examples of Spanning-Tree Protocol Configurations	255
Chapter 21	Summary of Spanning-Tree Protocol Configuration Statements	259
Part 7	Indexes	
	Index	299
	Index of Statements and Commands	309

Table of Contents

	About This Guide	xxi
	Junos Documentation and Release Notes	xxi
	Objectives	xxii
	Audience	xxii
	Supported Routing Platforms	xxiii
	Using the Indexes	xxiii
	Using the Examples in This Manual	xxiii
	Merging a Full Example	xxiii
	Merging a Snippet	xxiv
	Documentation Conventions	xxiv
	Documentation Feedback	xxvi
	Requesting Technical Support	xxvi
	Self-Help Online Tools and Resources	xxvii
	Opening a Case with JTAC	xxvii
Part 1	Overview	
Chapter 1	Layer 2 Services for MX Series Routers Overview	3
	MX Series Router Architecture	3
	MX Series Router Packet Forwarding and Data Flow	5
	Line Cards Supported on MX Series Routers	5
	FPCs and PICs	6
	DPCs	6
	Modular Port Concentrator (MPC) and Modular Interface Card (MIC) Interfaces	7
	Understanding Trio Layer 2 Feature Parity	7
	Features Operating Over Layer 2 but Configured at Layer 3	8
	Layer 2 and Layer 3 Features on MX Series Routers	8
	Multicast Snooping on MX Series Routers	8
	VPLS	9
	Layer 2 VPNs	9
	Ethernet Frame Counts and Statistics on MX Series Routers	10
Part 2	Configuration Basics for Layer 2 Services	
Chapter 2	Configuring Routing Instances for Layer 2 Services	13
	Routing Instances Overview	13
	Layer 2 Routing Instance Types	14
	Layer 2 Routing Instances Configuration Hierarchy	15
	Configuring a VPLS Routing Instance	16
	Configuring a Virtual Switch Routing Instance	17

	Configuring a Layer 2 Control Protocol Routing Instance	18
Chapter 3	Configuring Basic Layer 2 Services	21
	Multicast Snooping and VPLS Root Protection	21
	Multicast Snooping and VPLS Root Protection Overview	21
	Configuring Multicast Snooping to Ignore Spanning Tree Topology Change Messages	23
	Example: Configuring Multicast Snooping for a Bridge Domain	24
	Load Balancing and Ethernet Link Aggregation	25
	Load Balancing and Ethernet Link Aggregation Overview	25
	Configuring Load Balancing on a LAG Link	25
	Example: Configuring Load Balancing on a LAG Link	26
	Layer 2 Learning and Forwarding in a Logical System	26
	Layer 2 Learning and Forwarding in a Logical System Overview	27
	Enabling Layer 2 Learning and Forwarding in a Logical System	27
	Example: Configuring Layer 2 Learning and Forwarding and RSTP in a Logical System	28
Chapter 4	Configuring Link Layer Discovery Protocol	31
	LLDP Overview	31
	Configuring LLDP	32
	Tracing LLDP Operations	34
	Example: Configuring LLDP	35
Chapter 5	Summary of Link Layer Discovery Protocol Configuration Statements . . .	37
	advertisement-interval	37
	disable	38
	hold-multiplier	38
	interface	39
	lldp	40
	lldp-configuration-notification-interval	41
	ptopo-configuration-maximum-hold-time	41
	ptopo-configuration-trap-interval	42
	traceoptions	43
	transmit-delay	45
Chapter 6	Configuring Multiple VLAN Registration Protocol	47
	Understanding Multiple VLAN Registration Protocol (MVRP) on MX Series Routers	47
	How MVRP Works on MX Series Routers	48
	Basics of MVRP on MX Series Routers	48
	MVRP Registration Modes	48
	MRP Timers	49
	MRP VLAN Messages	49
	MVRP Limitations	49
	Example: Configuring Automatic VLAN Administration Using MVRP on MX Series Routers	50
	Configuring Multiple VLAN Registration Protocol (MVRP)	63
	Enabling MVRP	63
	Disabling MVRP	63

	Changing the Registration Mode to Disable Dynamic VLANs	63
	Configuring Timer Values	64
	Configuring the Multicast MAC address for MVRP	64
	Configuring an MVRP Interface as a Point-to-Point Interface	64
	Configuring MVRP Tracing Options	65
	Controlling the Management State of a VLAN in MVRP Configurations on MX Series Routers (CLI Procedure)	65
	Configure All VLANs to Operate in Normal State	66
	Configure VLANs to Operate With Mixed States (Fixed and Normal)	67
	Configure VLANs to Operate With Mixed States (Fixed, Normal, and Forbidden)	67
	Verifying That MVRP Is Working Correctly	67
Chapter 7	Summary of Multiple VLAN Registration Protocol Configuration Statements	71
	bpdu-destination-mac-address	71
	interface (MVRP)	72
	join-timer (MVRP)	73
	leaveall-timer (MVRP)	74
	leave-timer (MVRP)	75
	mvrp	76
	no-dynamic-vlan	77
	point-to-point (MVRP)	78
	registration	79
	traceoptions (MVRP)	80
Part 3	Layer 2 Port Mirroring	
Chapter 8	Layer 2 Port Mirroring Overview	85
	Layer 2 Port Mirroring Overview	85
	Layer 2 Port Mirroring Properties	86
	Packet-Selection Properties	86
	Packet Address Family	86
	Mirror Destination Properties	87
	Mirror-Once Option	87
	Layer 2 Port Mirroring Global Instance	87
	Layer 2 Port Mirroring Named Instances	88
	Layer 2 Port Mirroring Named Instances Overview	88
	Mirroring at Ports Grouped at the FPC Level	89
	Mirroring at Ports Grouped at the PIC Level	89
	Mirroring at a Group of Ports Bound to Multiple Named Instances	89
	Layer 2 Port Mirroring Firewall Filters	90
	Layer 2 Port Mirroring Firewall Filters Overview	90
	Mirroring of Packets Received or Sent on a Logical Interface	91
	Mirroring of Packets Forwarded or Flooded to a Bridge Domain	91
	Mirroring of Packets Forwarded or Flooded to a VPLS Routing Instance	91
	Layer 2 Port Mirroring to Multiple Destinations Using Next-Hop Groups	92
	Guidelines for Configuring Layer 2 Port Mirroring	93
	Restrictions on Layer 2 Port Mirroring	93
	Application of Layer 2 Port Mirroring Types	94

	Precedence of Multiple Levels of Layer 2 Port Mirroring on a Physical Interface	96
	Behavior of Layer 2 Port Mirroring of Logical Interfaces on PE Routers	96
	Layer 2 Port Mirroring of PE Router Logical Interfaces	96
	Layer 2 Port Mirroring of PE Router Aggregated Ethernet Interfaces	98
Chapter 9	Configuring Layer 2 Port Mirroring	101
	Configuring Layer 2 Port Mirroring for Physical Interfaces	101
	Configuring the Global Instance of Layer 2 Port Mirroring	101
	Defining a Named Instance of Layer 2 Port Mirroring	104
	Displaying Information About DPCs or FPCs in an MX Series Router	107
	Binding Layer 2 Port Mirroring to Ports Grouped at the FPC Level	108
	Binding Layer 2 Port Mirroring to Ports Grouped at the PIC Level	110
	Displaying Layer 2 Port-Mirroring Instance Settings and Status	111
	Disabling Layer 2 Port Mirroring Instances	112
	Configuring Layer 2 Port Mirroring for Logical Interfaces	113
	Defining a Layer 2 Port-Mirroring Firewall Filter	113
	Applying Layer 2 Port Mirroring to a Logical Interface	117
	Applying Layer 2 Port Mirroring to Traffic Forwarded or Flooded to a Bridge Domain	120
	Applying Layer 2 Port Mirroring to Traffic Forwarded or Flooded to a VPLS Routing Instance	122
	Configuring Layer 2 Port Mirroring to Multiple Destinations	124
	Defining a Next-Hop Group for Layer 2 Port Mirroring	124
	Displaying Next-Hop Group Settings and Status	125
Chapter 10	Examples of Layer 2 Port Mirroring	127
	Examples: Layer 2 Port-Mirroring at Multiple Levels of the Chassis	127
	Layer 2 Port Mirroring at the FPC Level	127
	Layer 2 Port Mirroring at the PIC Level	128
	Layer 2 Port Mirroring at the FPC and PIC Levels	128
	Example: Layer 2 Port Mirroring with Multiple Instances	129
	Example: Configuring Multiple Instances of Layer 2 Port Mirroring	129
	Explicit Reference of a Port Mirroring Instance	131
	Implicit Reference of Port Mirroring on the Underlying Physical Interface	132
	Example: Layer 2 Port Mirroring at a Logical Interface	133
	Example: Layer 2 Port Mirroring for a Layer 2 VPN	135
	Example: Layer 2 Port Mirroring for a Layer 2 VPN with LAG Links	138
	Example: Layer 2 Port Mirroring to Multiple Destinations	140
Part 4	Layer 2 Bridging	
Chapter 11	Layer 2 Bridging Overview	147
	Layer 2 Bridge Domains Overview	147
	Layer 2 Virtual Switches Overview	148
	Layer 2 Learning and Forwarding for Bridge Domains Overview	149
	Layer 2 Learning and Forwarding for Bridge Domains Functioning as Switches with Layer 2 Trunk Ports	149
	Guidelines for Configuring VLAN Identifiers for Bridge Domains and VPLS Routing Instances	150

Chapter 12	Configuring Layer 2 Bridging	151
	Configuring Bridge Domains for Layer 2 Bridging and Layer 3 IP Routing	151
	Configuring a Bridge Domain	151
	Configuring VLAN Identifiers for Bridge Domains and VPLS Routing Instances	152
	Configuring Integrated Routing and Bridging for Bridge Domains	158
	Configuring Bridge Domains as Switches for Layer 2 Trunk Ports	159
	Configuring Layer 2 Virtual Switches	160
	Configuring a Layer 2 Virtual Switch	160
	Configuring a Layer 2 Virtual Switch with a Layer 2 Trunk Port	162
	Configuring VPLS Ports in a Virtual Switch	163
	Configuring Integrated Routing and Bridging for a Bridge Domain in a Layer 2 Virtual Switch	164
	Configuring Layer 2 Learning and Forwarding for Bridge Domains	166
	Disabling MAC Learning for a Bridge Domain or Logical Interface	166
	Configuring Static MAC Addresses for Logical Interfaces in a Bridge Domain	167
	Configuring the Size of the MAC Address Table	168
	Limiting MAC Addresses Learned from an Interface in a Bridge Domain	168
	Enabling MAC Accounting for a Bridge Domain	170
	Configuring Layer 2 Learning and Forwarding for Bridge Domains Functioning as Switches with Layer 2 Trunk Ports	170
	Disabling MAC Learning for a Set of Bridge Domains	170
	Limiting the Number of MAC Addresses Learned from a Trunk Port	171
	Configuring the Size of the MAC Address Table for a Set of Bridge Domains	172
	Enabling MAC Accounting for a Set of Bridge Domains	172
	Configuring Layer 3 Tunnel Services Interfaces on an MX Series Router with a DPC	172
Chapter 13	Summary of Layer 2 Bridging Configuration Statements	175
	bandwidth	175
	bridge-domains	176
	bridge-options	177
	domain-type	178
	interface	178
	interface-mac-limit	179
	mac-statistics	180
	mac-table-size	181
	no-irb-layer-2-copy	182
	no-mac-learning	183
	packet-action	184
	routing-interface	185
	static-mac	186
	switch-options	187
	tunnel-services	188
	vlan-id (Bridge Domain)	189
	vlan-id-list	190
	vlan-tags	191

Part 5	Layer 2 Address Learning and Forwarding	
Chapter 14	Layer 2 Learning and Forwarding	195
	Layer 2 Learning and Forwarding Overview	195
	Configuring Layer 2 Learning and Forwarding	196
	Configuring the MAC Table Timeout Interval	196
	Enabling MAC Accounting	196
	Limiting the Number of MAC Addresses Learned from Each Logical Interface	197
	Disabling Layer 2 Learning and Forwarding	198
Chapter 15	Summary of Layer 2 Address Learning and Forwarding Configuration Statements	199
	global-mac-limit	199
	global-mac-statistics	200
	global-mac-table-aging-time	200
	global-no-mac-learning	201
	l2-learning	201
Part 6	Spanning-Tree Protocols	
Chapter 16	Spanning-Tree Protocols Overview	205
	Spanning-Tree Protocols Supported on MX Series Routers	205
	BPDU Overview	206
Chapter 17	Guidelines for Configuring Spanning-Tree Protocols	209
	Spanning-Tree Protocols in Logical Systems	209
	IEEE 802.1D STP Version Forced for RSTP or VSTP	209
	RSTP or VSTP Forced to Run as IEEE 802.1D STP	210
	Reverting RSTP or VSTP Back From Forced IEEE 802.1D STP	210
	Basic Configuration of Bridges in Spanning-Tree Instances	211
	Provider Bridge Participation in RSTP or MSTP Instances	211
	System Identifier for Bridges in STP or RSTP Instances	212
	Bridge Priority for Election of Root Bridge and Designated Bridge	212
	Maximum Age for Awaiting Arrival of Hello BPDUs	212
	Hello Time for Root Bridge to Transmit Hello BPDUs	212
	Forward Delay Before Ports Transition to Forwarding State	213
	Physical Interface Configuration for Spanning-Tree Protocol Instances	213
	Spanning-Tree Instance Interface	213
	Spanning-Tree Instance Interface Priority	214
	Spanning-Tree Instance Interface Cost	214
	Spanning-Tree Instance Interface Point-to-Point Link Mode	215
	Spanning-Tree Instance Interface Configured as an Edge Port	215
	Spanning-Tree Protocol Trace Options	216
Chapter 18	Configuring Spanning-Tree Protocols	217
	Configuring Rapid Spanning-Tree Protocol	217
	Configuring Multiple Spanning-Tree Protocol	220
	Configuring Multiple Spanning-Tree Protocol	220
	Configuring MST Instances on a Physical Interface	224

	Disabling MSTP	225
	Configuring VLAN Spanning-Tree Protocol	226
	Tracing Spanning-Tree Operations	230
Chapter 19	Spanning-Tree Protocol Options	233
	Loop Protection for Spanning-Tree Instance Interfaces	233
	Loop Protection for Spanning-Tree Instance Interfaces Overview	233
	Loop Protection for a Spanning-Tree Instance Interface	234
	Configuring Loop Protection for a Spanning-Tree Instance Interface	235
	Root Protect for Spanning-Tree Instance Interfaces	236
	Root Protection for Spanning-Tree Instance Interfaces Overview	236
	Root Protect for a Spanning-Tree Instance Interface	236
	Enabling Root Protect for a Spanning-Tree Instance Interface	237
	BPDU Protection for Spanning-Tree Instance Interfaces	237
	BPDU Protection for Spanning-Tree Instance Interfaces Overview	238
	BPDU Protection for Individual Spanning-Tree Instance Interfaces	238
	Configuring BPDU Protection on Individual Interfaces	239
	BPDU Protection on All Edge Ports of the Bridge	240
	Configuring BPDU Protection on All Edge Ports	240
	Checking the Status of Spanning-Tree Instance Interfaces	241
	Clearing the Blocked Status of a Spanning-Tree Instance Interface	241
	VPLS Root Protection Topology Change Actions	242
	VPLS Multihomed Layer 2 Ring and MPLS Infrastructure Overview	242
	VPLS Multihomed Layer 2 Ring and MPLS Infrastructure Topology	243
	VPLS Multihoming: Priority of the Backup Bridge	245
	VPLS Multihoming: Hold Time Before Switching to Primary Priority	245
	VPLS Multihoming: System Identifier for Bridges in the Ring	246
	VPLS Multihoming: Bridge Flush of MAC Cache on Topology Change	247
	Configuring VPLS Root Protection Topology Change Actions to Control	
	Global Spanning Tree Behavior	247
	Configuring VPLS Root Protection Topology Change Actions to Control	
	VLAN Spanning Tree Behavior	249
	Layer 2 Protocol Tunneling Through a Network	250
	Layer 2 Protocol Tunneling Through a Network Overview	250
	MAC Address Rewriting Enabled for Layer 2 Protocol Tunneling	251
	Layer 2 Protocol Tunnel Interface	251
	Layer 2 Protocol to be Tunneled	252
	Configuring Layer 2 Protocol Tunneling	253
	Checking for a MAC Rewrite Error Condition Blocking a Spanning-Tree	
	Instance Interface	254
	Clearing a MAC Rewrite Error Condition Blocking a Spanning-Tree Instance	
	Interface	254
Chapter 20	Examples of Spanning-Tree Protocol Configurations	255
	Example: Enabling Loop Protection for Spanning-Tree Protocols	255
	Example: Blocking BPDUs on Aggregated Ethernet for 600 Seconds	255
	Example: Configuring VPLS Root Topology Change Actions	256
	Example: Tracing Spanning-Tree Protocol Operations	256

Chapter 21	Summary of Spanning-Tree Protocol Configuration Statements	259
	backup-bridge-priority	260
	bpdu-block	261
	bpdu-block-on-edge	261
	bpdu-destination-mac-address (Spanning Tree)	262
	bpdu-timeout-action	263
	bridge-priority	264
	configuration-name	265
	cost	266
	disable	267
	disable-timeout	267
	edge	268
	extended-system-id	269
	force-version	269
	forward-delay	270
	hello-time	271
	interface	272
	interface (BPDU Blocking)	272
	interface (Layer 2 Protocol Tunneling)	272
	interface (Spanning Tree)	273
	layer2-control	274
	mac-rewrite	275
	max-age	275
	max-hops	276
	mode	277
	msti	278
	mstp	279
	no-root-port	280
	priority	281
	priority-hold-time	282
	protocol	283
	protocols	284
	revision-level	285
	rstp	286
	system-id	287
	traceoptions (Spanning Tree)	288
	vlan	291
	vlan (MSTP)	291
	vlan (VSTP)	292
	vpls-flush-on-topology-change	293
	vstp	294
Part 7	Indexes	
	Index	299
	Index of Statements and Commands	309

List of Figures

Part 1	Overview	
Chapter 1	Layer 2 Services for MX Series Routers Overview	3
	Figure 1: MX Series Router Packet Forwarding and Data Flow	5
Part 2	Configuration Basics for Layer 2 Services	
Chapter 6	Configuring Multiple VLAN Registration Protocol	47
	Figure 2: MVRP Configured on Three MX Series Routers for Automatic VLAN Administration	52
Part 6	Spanning-Tree Protocols	
Chapter 19	Spanning-Tree Protocol Options	233
	Figure 3: VPLS Multihoming Configuration	244

List of Tables

	About This Guide	xxi
	Table 1: Notice Icons	xxv
	Table 2: Text and Syntax Conventions	xxv
Part 1	Overview	
Chapter 1	Layer 2 Services for MX Series Routers Overview	3
	Table 3: Trio Layer 2 Feature Parity	7
Part 2	Configuration Basics for Layer 2 Services	
Chapter 6	Configuring Multiple VLAN Registration Protocol	47
	Table 4: Components of the Network Topology	52
	Table 5: MVRP Management States	66
Part 3	Layer 2 Port Mirroring	
Chapter 8	Layer 2 Port Mirroring Overview	85
	Table 6: Application of Layer 2 Port Mirroring Types	94
	Table 7: Application of Layer 2 Port Mirroring Firewall Filters on PE Routers	97
Part 4	Layer 2 Bridging	
Chapter 12	Configuring Layer 2 Bridging	151
	Table 8: Statement Usage and Input Rewrite Operations for VLAN Identifiers for a Bridge Domain	156
	Table 9: Statement Usage and Output Rewrite Operations for VLAN Identifiers for a Bridge Domain	157

About This Guide

This preface provides the following guidelines for using the *Junos*[®] OS :

- Junos Documentation and Release Notes on page xxi
- Objectives on page xxii
- Audience on page xxii
- Supported Routing Platforms on page xxiii
- Using the Indexes on page xxiii
- Using the Examples in This Manual on page xxiii
- Documentation Conventions on page xxiv
- Documentation Feedback on page xxvi
- Requesting Technical Support on page xxvi

Junos Documentation and Release Notes

For a list of related Junos documentation, see
<http://www.juniper.net/techpubs/software/junos/> .

If the information in the latest release notes differs from the information in the documentation, follow the *Junos 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/> .

Juniper Networks supports a technical book program to publish books by Juniper Networks engineers and subject matter experts with book publishers around the world. These books go beyond the technical documentation to explore the nuances of network architecture, deployment, and administration using the Junos operating system (Junos OS) and Juniper Networks devices. In addition, the Juniper Networks Technical Library, published in conjunction with O'Reilly Media, explores improving network security, reliability, and availability using Junos OS configuration techniques. All the books are for sale at technical bookstores and book outlets around the world. The current list can be viewed at <http://www.juniper.net/books> .

Objectives

This guide provides an overview of the Layer 2 features supported in this release of Junos OS and describes how to configure the features to provide solutions to several network scenarios.



NOTE: For additional information about the Junos OS—either corrections to or information that might have been omitted from this guide—see the software release notes at <http://www.juniper.net/>.

Audience

This guide is designed for network administrators who are configuring and monitoring Layer 2 features.

To use this guide, you need a broad understanding of networks in general, the Internet in particular, networking principles, and network configuration. You must also be familiar with one or more of the following Internet routing protocols:

- Border Gateway Protocol (BGP)
- Distance Vector Multicast Routing Protocol (DVMRP)
- Intermediate System-to-Intermediate System (IS-IS)
- Internet Control Message Protocol (ICMP) router discovery
- Internet Group Management Protocol (IGMP)
- Multiprotocol Label Switching (MPLS)
- Open Shortest Path First (OSPF)
- Protocol-Independent Multicast (PIM)
- Resource Reservation Protocol (RSVP)
- Routing Information Protocol (RIP)
- Simple Network Management Protocol (SNMP)

Personnel operating the equipment must be trained and competent; must not conduct themselves in a careless, willfully negligent, or hostile manner; and must abide by the instructions provided by the documentation.

Supported Routing Platforms

For the Layer 2 features described in this manual, the Junos OS currently supports the following routing platforms:

- Juniper Networks MX Series 3D Universal Edge Routers

Using the Indexes

This reference contains a standard index with topic entries.

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

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

Merging a Full Example

To merge a full example, follow these steps:

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

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

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

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

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

Merging a Snippet

To merge a snippet, follow these steps:

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

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

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

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

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

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

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

For more information about the **load** command, see the [Junos OS CLI User Guide](#).

Documentation Conventions

Table 1 on page xxv 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.

Table 2 on page xxv 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
Fixed-width text like this	Represents output that appears on the terminal screen.	user@host> show chassis alarms No alarms currently active
<i>Italic text like this</i>	<ul style="list-style-type: none"> Introduces important new terms. Identifies book 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 System Basics Configuration 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.	Configure the machine's domain name: [edit] root@# set system domain-name <i>domain-name</i>
Text like this	Represents names of configuration statements, commands, files, and directories; interface names; 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)	Enclose optional keywords or variables.	stub <default-metric <i>metric</i> >;

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

Convention	Description	Examples
(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 <i>(string1 string2 string3)</i>
# (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)	Enclose a variable for which you can substitute one or more values.	community name members [community-ids]
Indentation and braces ({ })	Identify a level in the configuration hierarchy.	[edit] routing-options { static { route default { nexthop <i>address</i> ; retain; } } }
;(semicolon)	Identifies a leaf statement at a configuration hierarchy level.	
J-Web GUI Conventions		
Bold text like this	Represents J-Web 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.
> (bold right angle bracket)	Separates levels in a hierarchy of J-Web 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 send your comments to techpubs-comments@juniper.net, or fill out the documentation feedback form at <https://www.juniper.net/cgi-bin/docbugreport/>. If you are using e-mail, be sure to include the following information with your comments:

- Document or topic name
- URL or page number
- Software release version (if applicable)

Requesting Technical Support

Technical product support is available through the Juniper Networks Technical Assistance Center (JTAC). If you are a customer with an active J-Care or JNASC support contract,

or are covered under warranty, and need postsales 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/>
- 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: <https://www.juniper.net/alerts/>
- 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, visit us at <http://www.juniper.net/support/requesting-support.html>

PART 1

Overview

- [Layer 2 Services for MX Series Routers Overview on page 3](#)

CHAPTER 1

Layer 2 Services for MX Series Routers Overview

- MX Series Router Architecture on page 3
- MX Series Router Packet Forwarding and Data Flow on page 5
- Line Cards Supported on MX Series Routers on page 5
- Understanding Trio Layer 2 Feature Parity on page 7
- Features Operating Over Layer 2 but Configured at Layer 3 on page 8
- Ethernet Frame Counts and Statistics on MX Series Routers on page 10

MX Series Router Architecture

The key components of the Juniper Networks MX Series 3D Universal Edge Routers are the Dense Port Concentrators (DPCs), Modular Port Concentrators/Modular Interface Cards (MPCs/MICs), the Routing Engine (RE), and the Switch Control Board (SCB).

The DPCs are optimized for Ethernet density and are capable of supporting up to 40 Gigabit Ethernet or 4 10-Gigabit Ethernet ports. The DPC assembly combines packet forwarding and Ethernet interfaces on a single board, with four 10-Gbps Packet Forwarding Engines. Each Packet Forwarding Engine consists of one chip for Layer 3 processing and one Layer 2 network processor. The DPCs interface with the power supplies and Switch Control Boards.

Designed for flexibility, MPCs leverage the Junos Trio chipset to deliver the industry's highest density GbE and 10GbE, as well as the flexibility of modular interfaces, across the MX Series portfolio. These advanced capabilities allow mix and match interfaces to create service-specific and "pay as you grow" configurations. The MPC houses the PFEs to deliver up to 120 Gbps of comprehensive Layer 3 routing (IPv4 and IPv6), and Layer 2 switching. These MPCs also support inline services and advanced Hierarchical QoS (H-QoS) per MX Series slot.

Modular Interface Cards (MICs) install into Modular Port Concentrators (MPCs) and provide the physical connections to various network media types. MICs allow different physical interfaces to be supported on a single line card. You can install MICs of different media types on the same router as long as the router supports those MICs. MICs receive incoming packets from the network and transmit outgoing packets to the network. During this process, each MIC performs framing and high-speed signaling for its media type.

Before transmitting outgoing data packets through the MIC interfaces, the MPCs encapsulate the packets received. MICs are hot-removable and hot-insertable. You can install up to two MICs in the slots in each MPC.

The RE provides control plane functions and runs Junos OS. Software processes that run on the RE maintain the routing tables, manage the routing protocols used on the router, control the router interfaces, control some chassis components, and provide the interface for system management and user access to the router. REs communicate with DPCs and MPCs via dedicated out-of-band management channels, providing a clear distinction between the control and forwarding planes. Integrated into the SCB is the switch fabric, which interconnects all of the DPCs and MPCs within the chassis. The RE installs directly into the SCB.

The SCB powers cards on and off; controls clocking, resets and booting; and monitors and controls systems functions, including fan speed, board power status, Power Distribution Module (PDM) status and control, and the system front panel. Integrated into the SCB is the switch fabric, which interconnects all of the DPCs and MPCs within the chassis, supporting up to 48 Packet Forwarding Engines. The Routing Engine installs directly into the SCB.



NOTE: The MX80 3D Universal Edge Router leverages the technology used in the MPCs, common across the MX Series, and can accommodate multiple combinations of Modular Interface Cards (MICs) for increased flexibility. The MX80 is a single board router with a built-in RE and one Packet Forwarding Engine (PFE). The PFE has two “pseudo” Flexible PIC Concentrators (FPC 0 and FPC 1). Because there is no switching fabric, the single PFE takes care of both ingress and egress packet forwarding.

The MX Series router has been optimized for Ethernet services. Examples of the wide range of Ethernet services provided by the MX Series include:

- Virtual private LAN service (VPLS) for multipoint connectivity—Native support for VPLS services
- Virtual leased line (VLL) for point-to-point services—Native support for point-to-point services
- RFC 2547.bis IP/MPLS VPN (L3VPN)—Full support for MPLS VPNs throughout the Ethernet network
- Video distribution IPTV services
- Ethernet aggregation at the campus/enterprise edge—Supports dense 1-Gigabit Ethernet and 10-Gigabit Ethernet configurations, and provides full Layer 3 support for campus edge requirements
- Ethernet aggregation at the multiservice edge—Supports up to 480 1-Gigabit Ethernet ports or 48 10-Gigabit Ethernet ports for maximum Ethernet density along, with full Layer 2 and Layer 3 VPN support for MSE applications

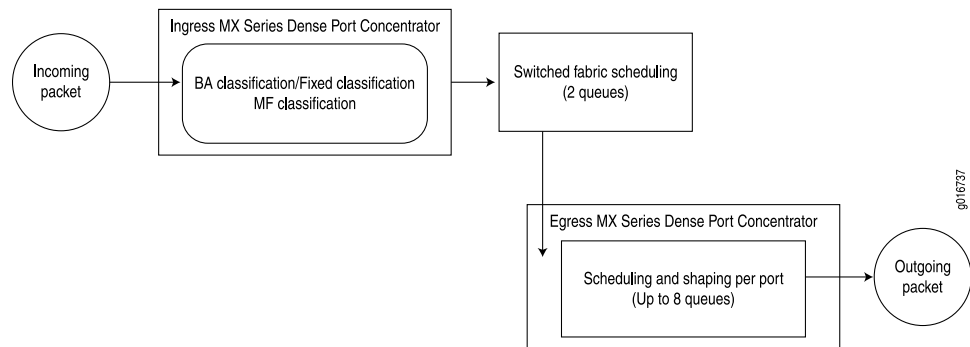
- Related Documentation**
- MX Series Router Packet Forwarding and Data Flow on page 5
 - Line Cards Supported on MX Series Routers on page 5
 - Layer 2 and Layer 3 Features on MX Series Routers on page 8
 - Ethernet Frame Counts and Statistics on MX Series Routers on page 10

MX Series Router Packet Forwarding and Data Flow

The architecture for Juniper Networks MX Series 3D Universal Edge Routers such as the MX960 Ethernet Services Router is similar in concept, but different in particulars, from other routing platforms.

The general architecture for the MX Series router is shown in Figure 1 on page 5.

Figure 1: MX Series Router Packet Forwarding and Data Flow



- Related Documentation**
- MX Series Router Architecture on page 3
 - Line Cards Supported on MX Series Routers on page 5
 - Layer 2 and Layer 3 Features on MX Series Routers on page 8
 - Ethernet Frame Counts and Statistics on MX Series Routers on page 10

Line Cards Supported on MX Series Routers

Juniper Networks MX Series 3D Universal Edge Routers process incoming and outgoing packets on several different types of line cards, including Dense Port Concentrators (DPCs), Flexible Port Concentrators (FPCs) with associated Physical Interface Cards (PICs), Trio Modular Port Concentrators (MPCs) with associated Modular Interface Cards (MICs). FPCs are populated with PICs for various interface types. DPCs and MPCs combine the functions of FPCs and the PICs, and with associated physical interfaces support a

variety of interface types. The configuration syntax for each type of line card is the same: ***type-fpc/pic/port***.

- FPCs and PICs on page 6
- DPCs on page 6
- Modular Port Concentrator (MPC) and Modular Interface Card (MIC) Interfaces on page 7

FPCs and PICs

An FPC occupies two slots when installed in an MX Series router. The maximum number of supported FPCs varies per router:

- MX960 router—6 FPCs
- MX480 router—3 FPCs
- MX240 router—1 FPC

PICs provide the physical connection to various network media types. The PICs are inserted into a slot in a router. You can install PICs of different media types on the same router as long as the router supports those PICs.

MX Series 3D Universal Edge Routers support 2 PICs per Flexible PIC Concentrator (FPC). The maximum number of supported PICs varies per router:

- MX960 router—12 PICs
- MX480 router—6 PICs
- MX240 router—2 PICs

DPCs

A DPC provides multiple physical interfaces and Packet Forwarding Engines on a single board that installs into a slot within the MX Series 3D Universal Edge Routers. The maximum number of supported DPCs varies per router:

- MX960 router—12 DPC slots
- MX480 router—6 DPC slots
- MX240 router—4 DPC slots



NOTE: In the Junos OS CLI, you use the FPC syntax to configure or display information about DPCs, and you use the PIC syntax to configure or display information about Packet Forwarding Engines on the DPCs.

In addition to Layer 3 routing capabilities, the DPCs also have many Layer 2 functions that allow MX Series routers to be used for many virtual LAN (VLAN) and other Layer 2 network applications.

Modular Port Concentrator (MPC) and Modular Interface Card (MIC) Interfaces

A Modular Port Concentrator supports two Modular Interface Card (MIC) interfaces. The maximum number of supported MPCs varies per router:

- MX960 router—12 MPC slots
- MX480 router—6 MPC slots
- MX240 router—3 MPC slots
- MX80 router—One fixed 10-Gigabit Ethernet MIC with four ports for uplink connections.



NOTE: The MX80 router is available as a modular (MX80) or fixed (MX80-48T) chassis. Both chassis have a fixed Modular Interface Card (MIC) that has 4 10-Gigabit Ethernet ports. The fixed MX80 router has an additional 48 10/100/1000Base-T RJ45 ports. The modular chassis has two dedicated slots for MICs.

Related Documentation

- MX Series Router Architecture on page 3
- MX Series Router Packet Forwarding and Data Flow on page 5
- Layer 2 and Layer 3 Features on MX Series Routers on page 8
- Ethernet Frame Counts and Statistics on MX Series Routers on page 10

Understanding Trio Layer 2 Feature Parity

A variety of Layer 2 features are supported on M Series and MX Series routers. The features supported by the Trio family of line cards are listed in Table 3 on page 7.

Table 3: Trio Layer 2 Feature Parity

Feature	Feature Parity with Junos OS Release	Feature Supported in Junos OS Release
MX routers only: load balancing enhancements for Layer 2 Link Aggregation	9.1R1	10.4R1
Ethernet OAM IEEE 802.1ag MIP support	9.1R1	10.4R1
Link Layer Discovery Protocol (LLDP)	9.1R1	10.4R1
MX routers only: BPDU guard	9.1R1	10.4R1
MX routers only: BPDU loop guard	9.1R1	10.4R1
For next generation VPNs: IRB support with LDP-VLPS and BGP-VPLS interworking	9.1R1	10.4R1
MPLS: BGP multihoming for inter-AS VPLS	9.1R1	10.4R1

Table 3: Trio Layer 2 Feature Parity (*continued*)

Feature	Feature Parity with Junos OS Release	Feature Supported in Junos OS Release
MX routers only: Ethernet as a core-facing interface in VPLS	9.1R1	10.4R1
Disables next-hop flood in connectivity fault management (CFM)	9.1R1	10.4R1

Features Operating Over Layer 2 but Configured at Layer 3

The following topics describe features that operate over Layer 2 networks but are configured at Layer 3:

- Layer 2 and Layer 3 Features on MX Series Routers on page 8
- Multicast Snooping on MX Series Routers on page 8
- VPLS on page 9
- Layer 2 VPNs on page 9

Layer 2 and Layer 3 Features on MX Series Routers

You can configure MX Series routers to provide simultaneous support for Layer 2 and Layer 3 Ethernet services. In many cases, Layer 2 protocols run on some interfaces, and Layer 3 protocols run on others.

The [Junos OS Layer 2 Configuration Guide](#) discusses Layer 2 configurations on supported routers, including Layer 2 statement summaries and configuration statement examples. For more complete Layer 2 configuration examples for MX Series routers, see the [Junos OS MX Series 3D Universal Edge Routers Solutions Guide](#).

For more information about configuring Layer 3 features and functions (such as class of service), see the relevant Junos configuration guides.

Related Documentation

- MX Series Router Architecture on page 3
- MX Series Router Packet Forwarding and Data Flow on page 5
- Line Cards Supported on MX Series Routers on page 5
- Ethernet Frame Counts and Statistics on MX Series Routers on page 10

Multicast Snooping on MX Series Routers

Because MX Series routers can support both Layer 3 and Layer 2 functions at the same time, you can configure the Layer 3 multicast protocols Protocol Independent Multicast (PIM) and the Internet Group Membership Protocol (IGMP) as well as Layer 2 VLANs on an MX Series router.

Normal encapsulation rules restrict Layer 2 processing to accessing information in the frame header and Layer 3 processing to accessing information in the packet header. However, in some cases, an interface running a Layer 2 protocol needs information available only at Layer 3. In multicast applications, the VLANs need the group membership

information and multicast tree information available to the Layer 3 IGMP and PIM protocols. In these cases, the Layer 3 configurations can use PIM or IGMP snooping to provide the needed information at the VLAN level.

For information about configuring multicast snooping for the operational details of a Layer 3 protocol on behalf of a Layer 2 spanning-tree protocol process, see “Multicast Snooping and VPLS Root Protection Overview” on page 21.

Snooping configuration statements and examples are not included in the *Junos OS Layer 2 Configuration Guide*. For more information about configuring PIM and IGMP snooping, see the *Junos OS Multicast Protocols Configuration Guide*.

Related Documentation

- Multicast Snooping and VPLS Root Protection Overview on page 21
- Configuring Multicast Snooping to Ignore Spanning Tree Topology Change Messages on page 23
- Example: Configuring Multicast Snooping for a Bridge Domain on page 24

VPLS

In a Layer 3 network only, you can configure virtual private LAN service (VPLS), which is an Ethernet-based point-to-multipoint Layer 2 VPN. It enables you to connect geographically dispersed Ethernet local area networks (LAN) sites to each other across an MPLS backbone. For ISP customers who implement VPLS, all sites appear to be in the same Ethernet LAN even though traffic travels across the service provider's network.

For information about configuring VPLS, see the *Junos OS VPNs Configuration Guide*.

Related Documentation

- MX Series Router Architecture on page 3
- Layer 2 and Layer 3 Features on MX Series Routers on page 8

Layer 2 VPNs

In a Layer 3 network only, you can configure Layer 2 virtual private network (VPN) under a Layer 2 VPN routing instance type **l2vpn**.

In a Layer 2 environment, you can use a **l2vpn** routing instance to transparently carry Layer 2 traffic over an IP/MPLS backbone. Layer 2 traffic is sent to the provider edge (PE) router in Layer 2 format. The PE router encapsulates the frames and transports them over the IP/MPLS backbone to the PE router on the other side of the cloud. The remote PE router removes encapsulation and sends the frames to the receiving site in Layer 2 format.

For information about configuring an **l2vpn** routing instance and Layer 2 VPNs, see the *Junos OS VPNs Configuration Guide*.

For a detailed Layer 2 VPN example configuration, see the *Junos OS Feature Guides*.

For information about tunnel interfaces, see the *Junos OS Network Interfaces Configuration Guide*.

- Related Documentation**
- MX Series Router Architecture on page 3
 - Layer 2 and Layer 3 Features on MX Series Routers on page 8

Ethernet Frame Counts and Statistics on MX Series Routers

The following considerations apply to Ethernet frame counts and statistics on Juniper Networks MX Series 3D Universal Edge Routers:

- Interface counters *do not* include the 7-byte Ethernet frame preamble and the frame delimiter byte.
- In Media Access Control (MAC) statistics, the frame size includes the MAC header and cyclical redundancy check (CRC) *before* any VLAN rewrite or other rules are applied.
- In traffic statistics, the frame size includes the Layer 2 header without the trailer CRC and *after* any VLAN rewrite or other rules are applied.

- Related Documentation**
- MX Series Router Architecture on page 3
 - MX Series Router Packet Forwarding and Data Flow on page 5
 - Line Cards Supported on MX Series Routers on page 5
 - Layer 2 and Layer 3 Features on MX Series Routers on page 8

PART 2

Configuration Basics for Layer 2 Services

- [Configuring Routing Instances for Layer 2 Services on page 13](#)
- [Configuring Basic Layer 2 Services on page 21](#)
- [Configuring Link Layer Discovery Protocol on page 31](#)
- [Summary of Link Layer Discovery Protocol Configuration Statements on page 37](#)
- [Configuring Multiple VLAN Registration Protocol on page 47](#)
- [Summary of Multiple VLAN Registration Protocol Configuration Statements on page 71](#)

CHAPTER 2

Configuring Routing Instances for Layer 2 Services

- Routing Instances Overview on page 13
- Layer 2 Routing Instance Types on page 14
- Layer 2 Routing Instances Configuration Hierarchy on page 15
- Configuring a VPLS Routing Instance on page 16
- Configuring a Virtual Switch Routing Instance on page 17
- Configuring a Layer 2 Control Protocol Routing Instance on page 18

Routing Instances Overview

A routing instance is a routing entity for a router. You can create multiple instances of BGP, IS-IS, OSPF, OSPFv3, RIP, and static routes. Each instance contains a routing table, applied routing policies, routing table group, interfaces that belong to that instance, and a protocol-specific route configuration related to that instance.

You configure a primary routing instance at the **[edit protocols]** hierarchy level. You configure additional routing instances at the **[edit routing-instances]** or **[edit logical-systems *logical-system-name* routing-instance]** hierarchy level.

You use routing instances to:

- Create administrative separation in a large network to segregate customer traffic and associated settings. The customers see only the routes belonging to them.
- Create overlay networks in which separate services are routed only towards routers participating in that service, such as voice. The overlay network isolates routes belonging to one service from another service by exporting routes, applying tags, and filtering based on tags.

Each routing instance consists of sets of the following:

- A set of routing tables
- A set of interfaces that belong to these routing tables
- A set of routing option configurations

Each routing instance has a unique name and a corresponding IP unicast table. For example, if you configure a routing instance with the name **my-instance**, its corresponding IP unicast table will be **my-instance.inet.0**. All routes for **my-instance** are installed into **my-instance.inet.0**.

Routes are installed into the default routing instance **inet.0** by default, unless a routing instance is specified.

For details about configuring interfaces, see the [Junos OS Network Interfaces Configuration Guide](#).

Related Documentation

- Layer 2 Routing Instance Types on page 14
- Layer 2 Routing Instances Configuration Hierarchy on page 15
- Configuring a VPLS Routing Instance on page 16
- Configuring a Virtual Switch Routing Instance on page 17
- Configuring a Layer 2 Control Protocol Routing Instance on page 18

Layer 2 Routing Instance Types

Although routing instances are primarily intended to maintain separation of tables and protocols at Layer 3 (mirroring the traditional IP network separation at Layer 3), many aspects of routing instances make them convenient to use for Layer 2 applications and architectures as well. In Layer 2 applications, routing instances still help to maintain table, interface, and customer insulation, but with regard to media access control (MAC) addresses and VLAN tags as much as IP addresses.

You can configure three types of routing instances in Layer 2 networks on MX Series routers, as described in the indicated sections:

- **layer2-control** (MX Series routers only)—Layer 2 control protocol routing instance. For configuration information, see “Configuring Layer 2 Protocol Tunneling” on page 253, “Configuring BPDU Protection on Individual Interfaces” on page 239, and “Configuring BPDU Protection on All Edge Ports” on page 240.
- **virtual-switch** (MX Series routers only)—Virtual switch routing instance. For configuration information, see “Configuring a Layer 2 Virtual Switch” on page 160.
- **vpls**—Virtual private LAN service (VPLS) routing instance. For configuration information, see “Configuring a Bridge Domain” on page 151.

The other five types of routing instances are configured only for Layer 3 networks, and are described in the indicated Junos configuration guide:

- **forwarding**—Forwarding instance. For more information, see the [Junos OS Routing Protocols Configuration Guide](#).
- **l2vpn**—Layer 2 VPN routing instance. For more information, see the [Junos OS VPNs Configuration Guide](#).

- **no-forwarding**—Nonforwarding routing instance. For more information, see the [Junos OS Routing Protocols Configuration Guide](#).
- **virtual-router**—Virtual routing instance. For more information, see the [Junos OS Routing Protocols Configuration Guide](#).
- **vrf**—VPN routing and forwarding (VRF) instance. For more information, see the [Junos OS Routing Protocols Configuration Guide](#).

Related Documentation

- Routing Instances Overview on page 13
- Layer 2 Routing Instances Configuration Hierarchy on page 15
- Configuring a VPLS Routing Instance on page 16
- Configuring a Virtual Switch Routing Instance on page 17
- Configuring a Layer 2 Control Protocol Routing Instance on page 18

Layer 2 Routing Instances Configuration Hierarchy

Use the **vpls** routing instance type for point-to-multipoint LAN implementations between a set of sites in a VPN.

To configure routing instances for Layer 2 networks, include the following statements:

```
routing-instances {
  routing-instance-name {
    description text;
    forwarding-options {
      ...forwarding-options...
    }
    instance-type (layer2-control | virtual-switch | vpls);
    interface interface-name;
    no-vrf-advertise;
    route-distinguisher (as-number:number | ip-address:number);
    vrf-export [ policy-names ];
    vrf-import [ policy-names ];
    vrf-table-label;
    vrf-target {
      export community-name;
      import community-name;
    }
    protocols {
      ...protocols-configuration ...
    }
    routing-options {
      ...routing-options-configuration ...
    }
    bridge-domains {
      bridge-domain-name {
        domain-type bridge;
        interface interface-name;
        routing-interface routing-interface-name;
        vlan-id (Bridge Domain) (none | all | number);
      }
    }
  }
}
```

```
    vlan-tags outer number inner number;  
    bridge-options {  
        interface-mac-limit limit {  
            packet-action drop;  
        }  
        interface interface-name {  
            interface-mac-limit limit {  
                packet-action drop;  
            }  
        }  
        mac-statistics;  
        mac-table-size limit {  
            packet-action drop;  
        }  
        no-mac-learning;  
        static-mac mac-address;  
    }  
}  
}
```

With the exception of the **instance-type virtual-switch** statement (which configures a virtual-switch routing instance), you can include the statements at the following hierarchy levels:

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

The **instance-type virtual-switch** statement is not supported at the [edit logical-systems *logical-system-name*] hierarchy level.

Related Documentation

- Routing Instances Overview on page 13
- Layer 2 Routing Instance Types on page 14
- Configuring a VPLS Routing Instance on page 16
- Configuring a Virtual Switch Routing Instance on page 17
- Configuring a Layer 2 Control Protocol Routing Instance on page 18

Configuring a VPLS Routing Instance

Use the **vpls** routing instance type for point-to-multipoint LAN implementations between a set of sites in a VPN.

To create a routing instance for VPLS, include at least the following statements in the configuration:

```
routing-instances {  
    routing-instance-name {  
        instance-type vpls;  
        interface interface-name;  
        route-distinguisher (as-number:number | ip-address:number);  
    }  
}
```

```

vrf-import [ policy-names ];
vrf-export [ policy-names ];
protocols {
  vpls {
    ... vpls configuration ...
  }
}

```

You can include these statements at the following hierarchy levels:

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

For more information about configuring VPLS, see the [Junos OS VPNs Configuration Guide](#).
For a detailed VPLS example configuration, see the [Junos OS Feature Guides](#).

Related Documentation

- Routing Instances Overview on page 13
- Layer 2 Routing Instance Types on page 14
- Layer 2 Routing Instances Configuration Hierarchy on page 15
- Configuring a Virtual Switch Routing Instance on page 17
- Configuring a Layer 2 Control Protocol Routing Instance on page 18

Configuring a Virtual Switch Routing Instance

On MX Series routers only, use the **virtual-switch** routing instance type to isolate a LAN segment with its spanning-tree instance and to separate its VLAN ID space. A bridge domain consists of a set of ports that share the same flooding or broadcast characteristics. Each virtual switch represents a Layer 2 network. You can optionally configure a virtual switch to support Integrated Routing and Bridging (IRB), which facilitates simultaneous Layer 2 bridging and Layer 3 IP routing on the same interface. You can also configure Layer 2 control protocols to provide loop resolution. Protocols supported include the Spanning-Tree Protocol (STP), Rapid Spanning-Tree Protocols (RSTP), Multiple Spanning-Tree Protocol (MSTP), and VLAN Spanning-Tree Protocol (VSTP).

To create a routing instance for a virtual switch, include at least the following statements in the configuration:

```

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

```

```
    }
    protocols {
      (rstp | mstp | vstp) {
        ...stp-configuration ...
      }
    }
  }
}
```

The **instance-type virtual-switch** statement is not supported at the **[edit logical-systems logical-system-name]** hierarchy level.

For more information about configuring virtual switches, see “Configuring a Layer 2 Virtual Switch” on page 160.

**Related
Documentation**

- Routing Instances Overview on page 13
- Layer 2 Routing Instance Types on page 14
- Layer 2 Routing Instances Configuration Hierarchy on page 15
- Configuring a VPLS Routing Instance on page 16
- Configuring a Layer 2 Control Protocol Routing Instance on page 18

Configuring a Layer 2 Control Protocol Routing Instance

On MX Series routers only, use the **layer2-control** routing instance type for Rapid Spanning-Tree Protocol (RSTP) or Multiple Spanning-Tree Protocol (MSTP) in customer edge interfaces of a VPLS routing instance. Layer 2 control protocols enable features such as Layer 2 protocol tunneling or nonstop bridging. This instance type cannot be used if the customer edge interface is multihomed to two provider edge interfaces. If the customer edge interface is multihomed to two provider edge interfaces, use the default bridge protocol data unit (BPDU) tunneling.

To create a routing instance for Layer 2 control protocols, include at least the following statements in the configuration:

```
routing-instances {
  routing-instance-name {
    instance-type layer2-control;
    interface interface-name;
    route-distinguisher (as-number:number | ip-address:number);
    vrf-import [ policy-names ];
    vrf-export [ policy-names ];
    protocols {
      mstp {
        ... interface options ...
        msti msti-id {
          ... MSTP MSTI configuration ...
        }
      }
    }
  }
}
```

You can include these statements at the following hierarchy levels:

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

**Related
Documentation**

- Routing Instances Overview on page 13
- Layer 2 Routing Instance Types on page 14
- Layer 2 Routing Instances Configuration Hierarchy on page 15
- Configuring a VPLS Routing Instance on page 16
- Configuring a Virtual Switch Routing Instance on page 17

CHAPTER 3

Configuring Basic Layer 2 Services

- Multicast Snooping and VPLS Root Protection on page 21
- Load Balancing and Ethernet Link Aggregation on page 25
- Layer 2 Learning and Forwarding in a Logical System on page 26

Multicast Snooping and VPLS Root Protection

The following sections describe the configuration of multicast snooping with VPLS root protection for spanning-tree protocols:

- Multicast Snooping and VPLS Root Protection Overview on page 21
- Configuring Multicast Snooping to Ignore Spanning Tree Topology Change Messages on page 23
- Example: Configuring Multicast Snooping for a Bridge Domain on page 24

Multicast Snooping and VPLS Root Protection Overview

Snooping occurs when a Layer 2 protocol such as a spanning-tree protocol is aware of the operational details of a Layer 3 protocol such as the Internet Group Management Protocol (IGMP) or other multicast protocol. Snooping is necessary when Layer 2 devices such as VLAN switches must be aware of Layer 3 information such as the media access control (MAC) addresses of members of a multicast group.

VPLS root protection is a spanning-tree protocol process in which only one interface in a multihomed environment is actively forwarding spanning-tree protocol frames. This protects the root of the spanning tree against bridging loops, but also prevents both devices in the multihomed topology from snooped information, such as IGMP membership reports.

For example, consider a collection of multicast-capable hosts connected to two customer edge (CE) routers (CE1 and CE2) which are connected to each other (a CE1–CE2 link is configured) and multihomed to two provider edge (PE) routers (PE1 and PE2, respectively). The active PE only receives forwarded spanning-tree protocol information on the active PE–CE link, due to root protection operation. As long as the CE1–CE2 link is operational, this is not a problem. However, if the link between CE1 and CE2 fails, and the other PE becomes the active spanning-tree protocol link, no multicast snooping information is available on the new active PE. The new active PE will not forward multicast traffic to the CE and the hosts serviced by this CE router.

The service outage is corrected once the hosts send new group membership IGMP reports to the CE routers. However, the service outage can be avoided if multicast snooping information is available to both PEs in spite of normal spanning-tree protocol root protection operation.



NOTE: You can configure multicast snooping to ignore messages about spanning tree topology changes for the virtual-switch routing-instance type only.

**Related
Documentation**

- VPLS Multihomed Layer 2 Ring and MPLS Infrastructure Overview on page 242 in the [Junos OS Layer 2 Configuration Guide](#)
- Multicast Snooping on MX Series Routers on page 8 in the [Junos OS Layer 2 Configuration Guide](#)
- Configuring Multicast Snooping to Ignore Spanning Tree Topology Change Messages on page 23 in the [Junos OS Layer 2 Configuration Guide](#)
- Example: Configuring Multicast Snooping for a Bridge Domain on page 24 in the [Junos OS Layer 2 Configuration Guide](#)
- [Junos OS Multicast Protocols Configuration Guide](#)

Configuring Multicast Snooping to Ignore Spanning Tree Topology Change Messages

You can configure the multicast snooping process for a virtual switch to ignore VPLS root protection topology change messages.

Before you begin, complete the following tasks:

1. Configure the spanning-tree protocol. For configuration details, see one of the following topics:
 - Configuring Rapid Spanning-Tree Protocol on page 217
 - Configuring Multiple Spanning-Tree Protocol on page 220
 - Configuring VLAN Spanning-Tree Protocol on page 226
2. Configure VPLS root protection. For configuration details, see one of the following topics:
 - Configuring VPLS Root Protection Topology Change Actions to Control Global Spanning Tree Behavior on page 247
 - Configuring VPLS Root Protection Topology Change Actions to Control VLAN Spanning Tree Behavior on page 249

To configure multicast snooping to ignore spanning tree topology change messages:

1. Configure a **virtual-switch** routing instance to isolate a LAN segment with its VSTP instance.

- a. Enable configuration of a virtual switch routing instance:

```
[edit]
user@host# edit routing-instances routing-instance-name
user@host# set instance-type virtual-switch
```

You can configure multicast snooping to ignore messages about spanning tree topology changes for the **virtual-switch** routing-instance type only.

- b. Enable configuration of a bridge domain:

```
[edit routing-instances routing-instance-name]
user@host# edit bridge-domains bridge-domain-name
user@host# set domain-type bridge
```

- c. Configure the logical interfaces for the bridge domain in the virtual switch:

```
[edit routing-instances routing-instance-name bridge-domains bridge-domain-name]
user@host# set interface interface-name
```

- d. Configure the VLAN identifiers for the bridge domain in the virtual switch. For detailed information, see “Configuring a Virtual Switch Routing Instance” on page 17.

2. Configure the multicast snooping process to ignore any spanning tree topology change messages sent to the virtual switch routing instance:

```
[edit routing-instances routing-instance-name bridge-domains bridge-domain-name]
user@host# set multicast-snooping-options ignore-stp-topology-change
```

3. Verify the configuration of multicast snooping for the virtual-switch routing instance to ignore spanning tree topology change messages:

```
[edit routing-instances routing-instance-name bridge-domains bridge-domain-name]
user@host# top
user@host# show routing-instances
```

```
routing-instance-name {
  instance-type virtual-switch;
  bridge-domains {
    bridge-domain-name {
      domain-type bridge {
        interface interface-name;
        ...VLAN-identifiers-configuration...
        multicast-snooping-options {
          ignore-stp-topology-change;
        }
      }
    }
  }
}
```

Related Documentation

- Multicast Snooping on MX Series Routers on page 8
- Multicast Snooping and VPLS Root Protection Overview on page 21
- Example: Configuring Multicast Snooping for a Bridge Domain on page 24

Example: Configuring Multicast Snooping for a Bridge Domain

This example configures the multicast snooping option for a bridge domain named **ignore-STP** in a virtual switch routing instance named **vs_routing_instance_multihomed_CEs**:

```
[edit]
routing-instances {
  vs_routing_instance_multihomed_CEs {
    instance-type virtual-switch;
    bridge-domains {
      bd_ignore_STP {
        multicast-snooping-options {
          ignore-stp-topology-change;
        }
      }
    }
  }
}
```



NOTE: This is not a complete router configuration.

- Related Documentation**
- Multicast Snooping on MX Series Routers on page 8
 - Multicast Snooping and VPLS Root Protection Overview on page 21
 - Configuring Multicast Snooping to Ignore Spanning Tree Topology Change Messages on page 23

Load Balancing and Ethernet Link Aggregation

The following sections describe load balancing and Ethernet link aggregation:

- Load Balancing and Ethernet Link Aggregation Overview on page 25
- Configuring Load Balancing on a LAG Link on page 25
- Example: Configuring Load Balancing on a LAG Link on page 26

Load Balancing and Ethernet Link Aggregation Overview

You can create a link aggregation group (LAG) for a group of Ethernet ports. Layer 2 bridging traffic is load balanced across the member links of this group, making the configuration attractive for congestion concerns as well as for redundancy. You can configure up to 480 LAG bundles on a Juniper Networks MX Series Ethernet Services Router. Each LAG bundle contains up to 16 links.

By default, the hash key mechanism to load-balance frames across LAG interfaces is based on Layer 2 fields (such as frame source and destination address) as well as the input logical interface (unit). No Layer 3 or Layer 4 fields are examined and are part of the default hash process, so the default is not optimized for Layer 2 switching (the frame source and destination MAC addresses are the same). In a Layer 2 switch, one link is overutilized and other links are underutilized.

- Related Documentation**
- Configuring Load Balancing on a LAG Link on page 25
 - Load Balancing on a LAG Link on page 26

Configuring Load Balancing on a LAG Link

You can configure the load balancing hash key for Layer 2 traffic to use fields in the Layer 3 and Layer 4 headers inside the frame payload for load-balancing purposes using the **payload** statement. You can configure the statement to look at **layer-3** (and **source-address-only** or **destination-address-only** packet header fields) or **layer-4** fields. You configure this statement at the **[edit forwarding-options hash-key family multiservice]** hierarchy level.

You can configure Layer 3 or Layer 4 options, or both. The **source-address-only** or **destination-address-only** options are mutually exclusive. The **layer-3-only** statement is not available on MX Series routers.

For more information about link aggregation group (LAG) configuration, see the [Junos OS Network Interfaces Configuration Guide](#).

- Related Documentation**
- Load Balancing and Ethernet Link Aggregation on page 25
 - Load Balancing on a LAG Link on page 26

Example: Configuring Load Balancing on a LAG Link

This example configures the load-balancing hash key to use the source Layer 3 IP address option and Layer 4 header fields as well as the source and destination MAC addresses for load balancing on a link aggregation group (LAG) link:

```
[edit]
forwarding-options {
  hash-key {
    family multiservice {
      source-mac;
      destination-mac;
      payload {
        ip {
          layer-3 {
            source-address-only;
          }
          layer-4;
        }
      }
    }
  }
}
```

- Related Documentation**
- Load Balancing and Ethernet Link Aggregation on page 25
 - Configuring Load Balancing on a LAG Link on page 25

Layer 2 Learning and Forwarding in a Logical System

The following sections describe Layer 2 learning and forwarding in a logical system:

- Layer 2 Learning and Forwarding in a Logical System Overview on page 27
- Enabling Layer 2 Learning and Forwarding in a Logical System on page 27
- Example: Configuring Layer 2 Learning and Forwarding and RSTP in a Logical System on page 28

Layer 2 Learning and Forwarding in a Logical System Overview

You can partition a single physical router into multiple logical devices called *logical systems* that perform independent routing tasks. Logical systems perform a subset of the actions of a physical router and have their own unique routing tables, interfaces, policies, and routing instances. On MX Series routers only, you can enable Layer 2 learning and forwarding in a logical system for bridge domains or other virtual-switch routing instances.

When enabling Layer 2 learning and forwarding in a logical system for bridge domains or other virtual-switch routing instances, the following guidelines apply:

- You can only configure 16 logical systems.
- Logging is performed for the entire device and not per logical system.
- You cannot restart Layer 2 learning for an individual logical system.

Related Documentation

- Enabling Layer 2 Learning and Forwarding in a Logical System on page 27
- Layer 2 Learning and Forwarding and RSTP in a Logical System on page 28

Enabling Layer 2 Learning and Forwarding in a Logical System

On MX Series router, you can enable Layer 2 learning and forwarding in a logical system for bridge domains or other virtual-switch routing instances.

Before you begin, configure the interfaces for the logical system.

To configure Layer 2 learning and forwarding in a logical system for bridge domains or other virtual-switch routing instances:

1. Enable configuration of a logical system:

```
[edit]
user@host# edit logical-systems logical-system-name
```

For detailed information about logical systems, see the [Junos OS Routing Protocols Configuration Guide](#).

2. Enable configuration of a virtual-switch routing instance:

```
[edit logical-systems logical-system-name]
user@host# edit routing-instances routing-instance-name
user@host# set instance-type virtual-switch
```

3. Configure the set of bridge-domains or other virtual-switch routing instances.

4. Configure Layer 2 learning and forwarding properties for a set of bridge domains:

- a. Enable configuration of Layer 2 learning and forwarding properties:

```
[edit logical-systems logical-system-name]  
user@host# edit switch-options
```

- b. Configure Layer 2 learning and forwarding properties. For more information, see “Layer 2 Learning and Forwarding for Bridge Domains Overview” on page 149.

5. Verify the configuration:

```
[edit logical-systems logical-system-name switch-options]  
user@host# top  
user@host# show logical-systems
```

```
logical-system-name {  
  interfaces {  
    ...interface-configurations...  
  }  
  routing-instances {  
    instance-type virtual-switch;  
  }  
  bridge-domains {  
    ...bridge-domain-configuration...  
  }  
  switch-options {  
    interface logical-interface-name {  
      ...layer-2-learning-and-forwarding-configuration...  
    }  
  }  
  protocols {  
    (rstp | mstp | vstp) {  
      interface interface-name;  
      ...spanning-tree-protocol-configuration...  
    }  
  }  
}
```

**Related
Documentation**

- Layer 2 Learning and Forwarding in a Logical System on page 27
- Layer 2 Learning and Forwarding and RSTP in a Logical System on page 28

Example: Configuring Layer 2 Learning and Forwarding and RSTP in a Logical System

The following example configures a logical system and routing instance with its own bridge domain (**bd1**), switch options, and spanning- tree protocol (**rstp**).

```
[edit]  
interfaces {  
  ge-5/0/1 {  
    flexible-vlan-tagging;  
  }  
}  
logical-systems {  
  logical-sys1 {
```



```

interfaces {
  ge-5/0/1 {
    unit 0 {
      family bridge {
        interface-mode trunk;
        vlan-id-list 1–5;
      }
    }
    unit 3 {
      family bridge {
        interface-mode trunk;
        vlan-id-list 11–15;
      }
    }
  }
  ge-5/0/2 {
    unit 0 {
      family bridge {
        interface-mode trunk;
        vlan-id-list 1–5;
      }
    }
  }
}
routing-instances {
  routing-inst-1 {
    interface ge-5/0/2;
    instance-type virtual-switch;
    bridge-domains {
      vlan-id 1;
    }
    protocols {
      rstp {
        interface ge-5/0/2;
      }
    }
  }
}
bridge-domains {
  bd-1 {
    vlan-id 1;
  }
}
switch-options {
  interface ge-5/0/1.3 {
    interface-mac-limit {
      1400;
      packet-action drop;
    }
  }
}
protocols {
  rstp {
    interface ge-5/0/1;
  }
}

```

```
}  
}
```



NOTE: This is not a complete router configuration.

**Related
Documentation**

- Layer 2 Learning and Forwarding in a Logical System on page 27
- Enabling Layer 2 Learning and Forwarding in a Logical System on page 27

CHAPTER 4

Configuring Link Layer Discovery Protocol

- LLDP Overview on page 31
- Configuring LLDP on page 32
- Tracing LLDP Operations on page 34
- Example: Configuring LLDP on page 35

LLDP Overview

The Link Layer Discovery Protocol (LLDP) is an industry-standard, vendor-neutral method to allow networked devices to advertise capabilities, identity, and other information onto a LAN. The Layer 2 protocol, detailed in IEEE 802.1AB-2005, replaces several proprietary protocols implemented by individual vendors for their equipment.

LLDP allows network devices that operate at the lower layers of a protocol stack (such as Layer 2 bridges and switches) to learn some of the capabilities and characteristics of LAN devices available to higher layer protocols, such as IP addresses. The information gathered through LLDP operation is stored in a network device and is queried with SNMP. Topology information can also be gathered from this database.

Some of the information that can be gathered by LLDP (only minimal information is mandatory) is:

- System name and description
- Port name and description
- VLAN name and identifier
- IP network management address
- Capabilities of the device (for example, switch, router, or server)
- MAC address and physical layer information
- Power information
- Link aggregation information

LLDP frames are sent at fixed intervals on each port that runs LLDP. LLDP protocol data units (LLDP PDUs) are sent inside Ethernet frames and identified by their destination Media Access Control (MAC) address (**01:80:C2:00:00:0E**) and Ethertype (**0x88CC**).

Mandatory information supplied by LLDP is chassis ID, port ID, and a time-to-live value for this information.

LLDP is a powerful way to allow Layer 2 devices to gather details about other network-attached devices.

Related Documentation

- Configuring LLDP on page 32
- Tracing LLDP Operations on page 34
- Example: Configuring LLDP on page 35

Configuring LLDP

You configure LLDP by including the **lldp** statement and associated parameters at the **[edit protocols]** hierarchy level. The complete set of LLDP statements follows:

```
lldp {
  advertisement-interval seconds;
  disable;
  hold-multiplier number;
  interface (all | interface-name) {
    disable;
  }
  lldp-configuration-notification-interval seconds;
  ptopo-configuration-maximum-hold-time seconds;
  ptopo-configuration-trap-interval seconds;
  traceoptions {
    file filename <files number> <size size> <world-readable | no-world-readable>;
    flag flag <flag-modifier> <disable>;
  }
  transmit-delay seconds
}
```

The following statements have default values:

- **advertisement-interval**—The default value is 30 seconds. The allowable range is from 5 through 32768 seconds.
- **hold-multiplier**—The default values is 4. The allowable range is from 2 through 10.
- **ptopo-configuration-maximum-hold-time**—The default value is 300 seconds. The allowable range is from 1 through 2147483647 seconds.
- **transmit-delay**—The default values is 2 seconds. The allowable range is from 1 through 8192 seconds.

The following statements must be explicitly configured:

- **lldp-configuration-notification-interval**—The allowable range is from 0 through 3600 seconds. There is no default value.
- **ptopo-configuration-trap-interval**—The allowable range is from 1 through 2147483647 seconds. There is no default value.

To disable LLDP on all or a particular interface, include the **interfaces** statement at the **[edit protocols lldp]** hierarchy level:

```
interface (all | interface-name) {
  disable;
}
```

To disable LLDP on all interfaces, use the **all** option. To disable LLDP on a particular interface, include the **disable** statement with the interface name.



NOTE: The **interface-name** must be the physical interface (for example, **ge-1/0/0**) and not a logical interface (unit).

The advertisement interval determines the frequency that an LLDP interface sends LLDP advertisement frames. The default value is 30 seconds. The allowable range is from 5 through 32768 seconds. You adjust this parameter by including the **advertisement-interval** statement at the **[edit protocols lldp]** hierarchy level.

The hold multiplier determines the multiplier to apply to the advertisement interval. The resulting value in seconds is used to cache learned LLDP information before discard. The default value is 4. When used with the default advertisement interval value of 30 seconds, this makes the default cache lifetime 120 seconds. The allowable range of the hold multiplier is from 2 through 10. You adjust this parameter by including the **hold-multiplier** statement at the **[edit protocols lldp]** hierarchy level.

The transmit delay determines the delay between any two consecutive LLDP advertisement frames. The default value is 2 seconds. The allowable range is from 1 through 8192 seconds. You adjust this parameter by including the **transmit-delay** statement at the **[edit protocols lldp]** hierarchy level.

The physical topology configuration maximum hold time determines the time interval for which an agent device maintains physical topology database entries. The default value is 300 seconds. The allowable range is from 1 through 2147483647 seconds. You adjust this parameter by including the **ptopo-configuration-maximum-hold-time** statement at the **[edit protocols lldp]** hierarchy level.

The LLDP configuration notification interval determines the period for which trap notifications are sent to the SNMP Master Agent when changes occur in the database of LLDP information. This capability is disabled by default. The allowable range is from 0 (disabled) through 3600 seconds. You adjust this parameter by including the **lldp-configuration-notification-interval** statement at the **[edit protocols lldp]** hierarchy level.

The physical topology configuration trap interval determines the period for which trap notifications are sent to the SNMP Master Agent when changes occur in the global physical topology statistics. This capability is disabled by default. The allowable range is from 0 (disabled) through 3600 seconds. The LLDP agent sends traps to the SNMP Master Agent if this interval has a value greater than 0 and there is any change during the **lldp-configuration-notification-interval** trap interval. You adjust this parameter by

including the **ptopo-configuration-trap-interval** statement at the **[edit protocols lldp]** hierarchy level.

- Related Documentation**
- LLDP Overview on page 31
 - Tracing LLDP Operations on page 34
 - Example: Configuring LLDP on page 35

Tracing LLDP Operations

To trace LLDP operational traffic, you can specify options in the global **traceoptions** statement included at the **[edit routing-options]** hierarchy level, and you can specify LLDP-specific options by including the **traceoptions** statement:

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

You can include this statement at the following hierarchy levels:

- **[edit protocols lldp]**
- **[edit routing-instances *routing-instance-name* protocols lldp]**

You can specify the following LLDP-specific options in the LLDP **traceoptions** statement:

- **all**—Trace all operations.
- **config**—Log configuration events.
- **interface**—Trace interface update events.
- **protocol**—Trace protocol information.
- **rtsock**—Trace real-time socket events.
- **vlan**—Trace VLAN update events.



NOTE: Use the trace flag **all** with caution. This flag may cause the CPU to become very busy.

For general information about tracing and global tracing options, see the statement summary for the global **traceoptions** statement in the [Junos OS Routing Protocols Configuration Guide](#).

- Related Documentation**
- LLDP Overview on page 31
 - Configuring LLDP on page 32
 - Example: Configuring LLDP on page 35

Example: Configuring LLDP

The following example configures LLDP on interface **ge-1/1/1** but disables LLDP on all other interfaces, explicitly configures the default values for all automatically enabled features, and configures a value of 30 seconds for the LLDP configuration notification interval and a value of 30 seconds for the physical topology trap interval.

```
[edit]
protocols {
  lldp {
    advertisement-interval 30;
    hold-multiplier 4;
    interface all {
      disable;
    }
    interface ge-1/1/1;
    lldp-configuration-notification-interval 30;
    ptopo-configuration-maximum-hold-time 300;
    ptopo-configuration-trap-interval 30;
    transmit-delay 2;
  }
}
```

You verify operation of LLDP with several show commands:

- **show lldp <detail>**
- **show lldp neighbors *interface-name***
- **show lldp statistics *interface-name***
- **show lldp local-information**
- **show lldp remote-global-statistics**

You can clear LLDP neighbor information or statistics globally or on an interface:

- **clear lldp neighbors *interface-name***
- **clear lldp statistics *interface-name***

You can display basic information about LLDP with the **show lldp detail** command:

```
user@host> show lldp detail
LLDP                : Enabled
Advertisement interval : 30 Second(s)
Transmit delay       : 2 Second(s)
Hold timer           : 4 Second(s)
Notification interval : 30 Second(s)
Config Trap Interval  : 300 Second(s)
Connection Hold timer : 60 Second(s)
```

Interface	LLDP	Neighbor count
ge-1/1/1	Enabled	0

LLDP basic TLVs supported:
Chassis identifier, Port identifier, Port description, System name, System

description, System capabilities, Management address.

LLDP 802 TLVs supported:

Link aggregation, Maximum frame size, MAC/PHY Configuration/Status, Port VLAN ID, Port VLAN name.

For more details about the output of these commands, see the [Junos OS Routing Protocols and Policies Command Reference](#).

**Related
Documentation**

- [LLDP Overview on page 31](#)
- [Configuring LLDP on page 32](#)
- [Tracing LLDP Operations on page 34](#)

CHAPTER 5

Summary of Link Layer Discovery Protocol Configuration Statements

The following sections explain each of the Link Layer Discovery Protocol (LLDP) configuration statements. The statements are organized alphabetically.

advertisement-interval

Syntax	advertisement-interval <i>seconds</i> ;
Hierarchy Level	[edit protocols <i>lldp</i>], [edit routing-instances <i>routing-instance-name</i> protocols <i>lldp</i>]
Release Information	Statement introduced in Junos OS Release 9.6.
Description	(MX Series routers only) Configure an interval for LLDP advertisement.
Options	<i>seconds</i> —Interval between LLDP advertisement. Default: 30 Range: 5 through 32768
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 LLDP on page 32

disable

Syntax	disable;
Hierarchy Level	[edit protocols lldp], [edit protocols lldp interface (all <i>interface-name</i>)], [edit routing-instances <i>routing-instance-name</i> protocols lldp], [edit routing-instances <i>routing-instance-name</i> protocols lldp interface (all <i>interface-name</i>)]
Release Information	Statement introduced in Junos OS Release 9.6.
Description	(MX Series routers only) Disable LLDP globally or on an 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">Configuring LLDP on page 32

hold-multiplier

Syntax	hold-multiplier <i>number</i> ;
Hierarchy Level	[edit protocols lldp], [edit routing-instances <i>routing-instance-name</i> protocols lldp]
Release Information	Statement introduced in Junos OS Release 9.6.
Description	(MX Series routers only) Configure a value for the LLDP hold multiplier.
Options	<i>number</i> —Advertisement interval multiplier for LLDP cache discard. Default: 4 (giving 120 second LLDP cache lifetime with other defaults) Range: 2 through 10
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 LLDP on page 32

interface

Syntax	interface (all <i>interface-name</i>) { disable; }
Hierarchy Level	[edit protocols lldp], [edit routing-instances <i>routing-instance-name</i> protocols lldp]
Release Information	Statement introduced in Junos OS Release 9.6.
Description	(MX Series routers only) Specify an LLDP interface.
Options	<i>interface-name</i> —A valid physical interface name.



NOTE: On MX Series routers, you run LLDP on a physical interface, such as ge-1/0/0, and not at the logical interface (unit) level.

all—Run LLDP on all interfaces.

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 LLDP on page 32

lldp

Syntax	<pre>lldp { advertisement-interval <i>seconds</i>; disable; hold-multiplier <i>number</i>; interface (all <i>interface-name</i>) { disable; } lldp-configuration-notification-interval <i>seconds</i>; ptopo-configuration-maximum-hold-time <i>seconds</i>; ptopo-configuration-trap-interval <i>seconds</i>; traceoptions { file <i>filename</i> <files <i>number</i>> <size <i>maximum-file-size</i>> <world-readable no-world-readable>; flag <i>flag</i> <disable>; } }</pre>
Hierarchy Level	[edit protocols], [edit routing-instances <i>routing-instance-name</i> protocols]
Release Information	Statement introduced in Junos OS Release 9.6.
Description	(MX Series routers only) Specify LLDP configuration 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">Configuring LLDP on page 32

lldp-configuration-notification-interval

Syntax	lldp-configuration-notification-interval <i>seconds</i> ;
Hierarchy Level	[edit protocols lldp], [edit routing-instances <i>routing-instance-name</i> protocols lldp]
Release Information	Statement introduced in Junos OS Release 9.6.
Description	(MX Series routers only) Configure a time for the period of SNMP trap notifications to the Master Agent to wait regarding changes in database information.
Options	seconds —Time for the period of SNMP trap notifications about the LLDP database. This feature is disabled by default. Range: 0 through 3600
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 LLDP on page 32

ptopo-configuration-maximum-hold-time

Syntax	ptopo-configuration-maximum-hold-time <i>seconds</i> ;
Hierarchy Level	[edit protocols lldp], [edit routing-instances <i>routing-instance-name</i> protocols lldp]
Release Information	Statement introduced in Junos OS Release 9.6.
Description	(MX Series routers only) Configure a time to maintain dynamic topology entries.
Options	seconds —Time to maintain interval dynamic topology entries. Default: 300 Range: 1 through 2147483647
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 LLDP on page 32

ptopo-configuration-trap-interval

Syntax	<code>ptopo-configuration-trap-interval seconds;</code>
Hierarchy Level	[edit protocols lldp], [edit routing-instances <i>routing-instance-name</i> protocols lldp]
Release Information	Statement introduced in Junos OS Release 9.6.
Description	(MX Series routers only) Configure a time for the period of SNMP trap notifications to the Master Agent to wait regarding changes in topology global statistics.
Options	seconds —Time for the period of SNMP trap notifications about global statistics. This feature is disabled by default. Range: 0 through 3600
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 LLDP on page 32

traceoptions

Syntax	<pre> traceoptions { file <i>filename</i> <files <i>number</i>> <size <i>maximum-file-size</i>> <world-readable no-world-readable>; flag <i>flag</i> <disable>; } </pre>
Hierarchy Level	[edit protocols lldp], [edit routing-instances <i>routing-instance-name</i> protocols lldp]
Release Information	Statement introduced in Junos OS Release 9.6.
Description	Set LLDP protocol-level tracing options.
Default	The default LLDP 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 spanning-tree protocol tracing output in the file /var/log/stp-log.</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 LLDP-specific tracing options:</p> <ul style="list-style-type: none"> • all—Trace all operations. • config—Log configuration events. • interface—Trace interface update events. • protocol—Trace protocol information. • rtsock—Trace socket events. • vlan—Trace vlan update events.

The following are the global tracing options:

- **all**—All tracing operations.
- **config-internal**—Trace configuration internals.
- **general**—Trace general events.
- **normal**—All normal events. This is the 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. This is the default. If you do not include this option, tracing output is appended to an existing trace file.

size maximum-file-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	<ul style="list-style-type: none">• Tracing LLDP Operations on page 34

transmit-delay

Syntax	<code>transmit-delay <i>seconds</i>;</code>
Hierarchy Level	[edit protocols <code>lldp</code>], [edit routing-instances <i>routing-instance-name</i> protocols <code>lldp</code>]
Release Information	Statement introduced in Junos OS Release 9.6.
Description	(MX Series routers only) Configure a delay between two successive LLDP advertisements.
Options	<i>seconds</i> —Delay between two successive LLDP advertisements. Default: 2 Range: 1 through 8192
Required Privilege Level	<code>routing</code> —To view this statement in the configuration. <code>routing-control</code> —To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">Configuring LLDP on page 32

CHAPTER 6

Configuring Multiple VLAN Registration Protocol

- Understanding Multiple VLAN Registration Protocol (MVRP) on MX Series Routers on page 47
- Example: Configuring Automatic VLAN Administration Using MVRP on MX Series Routers on page 50
- Configuring Multiple VLAN Registration Protocol (MVRP) on page 63
- Controlling the Management State of a VLAN in MVRP Configurations on MX Series Routers (CLI Procedure) on page 65
- Verifying That MVRP Is Working Correctly on page 67

Understanding Multiple VLAN Registration Protocol (MVRP) on MX Series Routers

You can configure Multiple VLAN Registration Protocol (MVRP) on Juniper Networks MX Series routers. The primary purpose of MVRP is to manage dynamic VLAN registration in switching networks. In managing dynamic VLAN registration, MVRP also prunes VLAN information.

MVRP is an Layer 2 application protocol of the Multiple Registration Protocol (MRP) and is defined in the IEEE 802.1ak standard. MRP and MVRP were designed by IEEE to perform the same functions as Generic Attribute Registration Protocol (GARP) and GARP VLAN Registration Protocol (GVRP) while overcoming some GARP and GVRP limitations, in particular limitations involving bandwidth usage and convergence time in large networks with large numbers of VLANs.

MVRP was created by IEEE as a replacement application for GVRP. MVRP and GVRP cannot be run concurrently to share VLAN information in a switching network.

This topic describes:

- How MVRP Works on MX Series Routers on page 48
- Basics of MVRP on MX Series Routers on page 48
- MVRP Registration Modes on page 48
- MRP Timers on page 49

- MRP VLAN Messages on page 49
- MVRP Limitations on page 49

How MVRP Works on MX Series Routers

The VLAN registration information sent by MVRP protocol data units (PDUs) includes the current VLANs membership—that is, which routers are members of which VLANs—and which router interfaces are in which VLAN. MVRP shares all information in the PDU with all routers participating in MVRP in the switching network.

MVRP stays synchronized using these PDUs. The routers in the network participating in MVRP receive these PDUs during state changes and update their MVRP states accordingly. MVRP timers dictate when PDUs can be sent and when routers receiving MVRP PDUs can update their MVRP information.

VLAN information is distributed as part of the MVRP message exchange process and can be used to dynamically create VLANs, which are VLANs created on one switch and propagated to other routers as part of the MVRP message exchange process. Dynamic VLAN creation using MVRP is enabled by default but can be disabled.

As part of ensuring that VLAN membership information is current, MVRP removes routers and interfaces from the VLAN information when they become unavailable. Pruning VLAN information has these benefits:

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

Basics of MVRP on MX Series Routers

MVRP is disabled by default on all MX Series routers. You can configure MVRP on MX Series router interfaces to participate in MVRP for the switching network. MVRP can only be enabled on trunk interfaces, and dynamic VLAN configuration through MVRP is enabled by default when MVRP is enabled.

MVRP Registration Modes

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

The following MVRP registration modes are configurable:

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

MRP Timers

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

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

- Join timer—Controls the interval for the next MVRP PDU transmit opportunity.
- Leave timer—Controls the period of time that an interface on the 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: Maintain default timer settings unless there is a compelling reason to change the settings. Modifying timers to inappropriate values might cause an imbalance in the operation of MVRP.

MRP VLAN Messages

MVRP uses MRP messages to register and declare MVRP states for a switch and to inform the switching network that a switch is leaving MVRP. These messages are communicated as part of the PDU to communicate the state of a particular switch interface on the switching network to the other switches in the network.

The following messages are communicated for MVRP:

- Empty—VLAN information is not being declared and is not registered.
- In—VLAN information is not being declared but is registered.
- JoinEmpty—VLAN information is being declared but not registered.
- JoinIn—VLAN information is being declared and is registered.
- Leave—VLAN information that was previously registered is being withdrawn.
- LeaveAll—All registrations will be de-registered. Participants that want to participate in MVRP will need to re-register.
- New—VLAN information is new and possibly not previously registered.

MVRP Limitations

The following limitations apply when configuring MVRP:

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

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

Related Documentation

- Example: Configuring Automatic VLAN Administration Using MVRP on MX Series Routers on page 50
- Configuring Multiple VLAN Registration Protocol (MVRP) on page 63
- Controlling the Management State of a VLAN in MVRP Configurations on MX Series Routers (CLI Procedure) on page 65
- Verifying That MVRP Is Working Correctly on page 67

Example: Configuring Automatic VLAN Administration Using MVRP on MX Series Routers

Multiple VLAN Registration Protocol (MVRP) is used in carrier Ethernet networks to dynamically share virtual LAN (VLAN) information and to automatically configure necessary VLAN information. Automatically configuring VLANs on ports based on the current network configuration ensures that a router does not send traffic to an interface on the network with an inactive VLAN. In this way, MVRP reduces network overhead by limiting the scope of broadcast, unknown unicast, and multicast (BUM) traffic to interested devices only. MVRP also provides for rapid healing of network failures without interrupting services to unaffected VLANs and improves convergence times.

MVRP is a Layer 2 network protocol based on the IEEE standard 802.1ak amendment to 802.1Q-2005, *Standard for Local and Metropolitan Area Networks Virtual Bridged Local Area Networks - Amendment 07: Multiple Registration Protocol*.

This example describes how to use MVRP to automate administration of VLAN membership changes within your network and to dynamically create VLANs:

- Requirements on page 50
- Overview and Topology on page 50
- Configuration on page 53
- Verification on page 59

Requirements

This example uses the following hardware and software components:

- Two MX Series routers acting as edge switches
- One MX Series router acting as aggregation switch
- Junos OS Release 10.1 or later for MX Series routers

Overview and Topology

VLANs are statically configured on access interfaces on MX Series routers acting as edge switches. The VLAN membership information is propagated to the MX Series router

acting as an aggregation switch at the core by enabling MVRP on two trunk interfaces—one connecting edge switch 1 (ES1) to aggregation switch 1 (AS1) and the other connecting ES2 to AS1. Enabling MVRP on the trunk interface of each MX Series router in your network ensures that the active VLAN information for the routers in the network is propagated to each router through the trunk interfaces (the default registration mode for MVRP).

MVRP ensures that the VLAN membership information on the trunk interface is updated as the edge switch's access interfaces become active or inactive.

You do not need to explicitly bind a VLAN to the trunk interface. When MVRP is enabled, the trunk interface advertises all the VLANs that are active (bound to access interfaces) on that switch. An MVRP-enabled trunk interface does not advertise VLANs that have been configured on the switch but that are not currently bound to an access interface. For example, ES1 in the topology does not forward traffic to inactive VLAN 300 on ES2.

Rapid Spanning Tree Protocol (RSTP) is also configured on the trunk interfaces to promote a loop-free topology.

This example shows a network with two customer sites, **site-1** and **site-2**, using VLANs 100, 200, and 300.

ES1 supports all three VLANs and all three VLANs are active and bound to interfaces that are connected to three customers at **site-1**:

- **ge-11/2/6**—Access port connecting customer3-site1, VLAN ID 100.
- **ge-11/2/7**—Access port connecting customer2-site1, VLAN ID 200.
- **ge-11/2/8**—Access port connecting customer1-site1, VLAN ID 300.
- **ge-11/3/0**—Trunk port connecting ES1 to AS1.

ES2 has been configured to support two VLANs, and both VLANs are active and bound to interfaces that are connect to two customers at **site-2**:

- **ge-0/1/1**—Access port connecting customer1-site2, VLAN ID 100.
- **ge-0/2/0**—Access port connecting customer2-site2, VLAN ID 200.
- **ge-0/0/5**—Trunk port connecting ES2 to AS1.

AS1 learns the VLANs dynamically using MVRP through the connection to the edge switches. AS1 has two trunk interfaces:

- **ge-3/3/0**—Connects the router to edge switch ES1 on interface **ge-11/3/0**.
- **ge-3/0/5**—Connects the router to edge switch ES2 on interface **ge-0/0/5**.

The default MVRP interface registration mode is **normal** and is used in this example. An interface in normal registration mode participates in MVRP when MVRP is enabled on the router. For information about changing the MVRP registration mode, refer to “Controlling the Management State of a VLAN in MVRP Configurations on MX Series Routers (CLI Procedure)” on page 65.

Figure 1 shows MVRP configured on three MX Series routers; two routers operating as edge switches and one router operating as an aggregation switch.

Figure 2: MVRP Configured on Three MX Series Routers for Automatic VLAN Administration

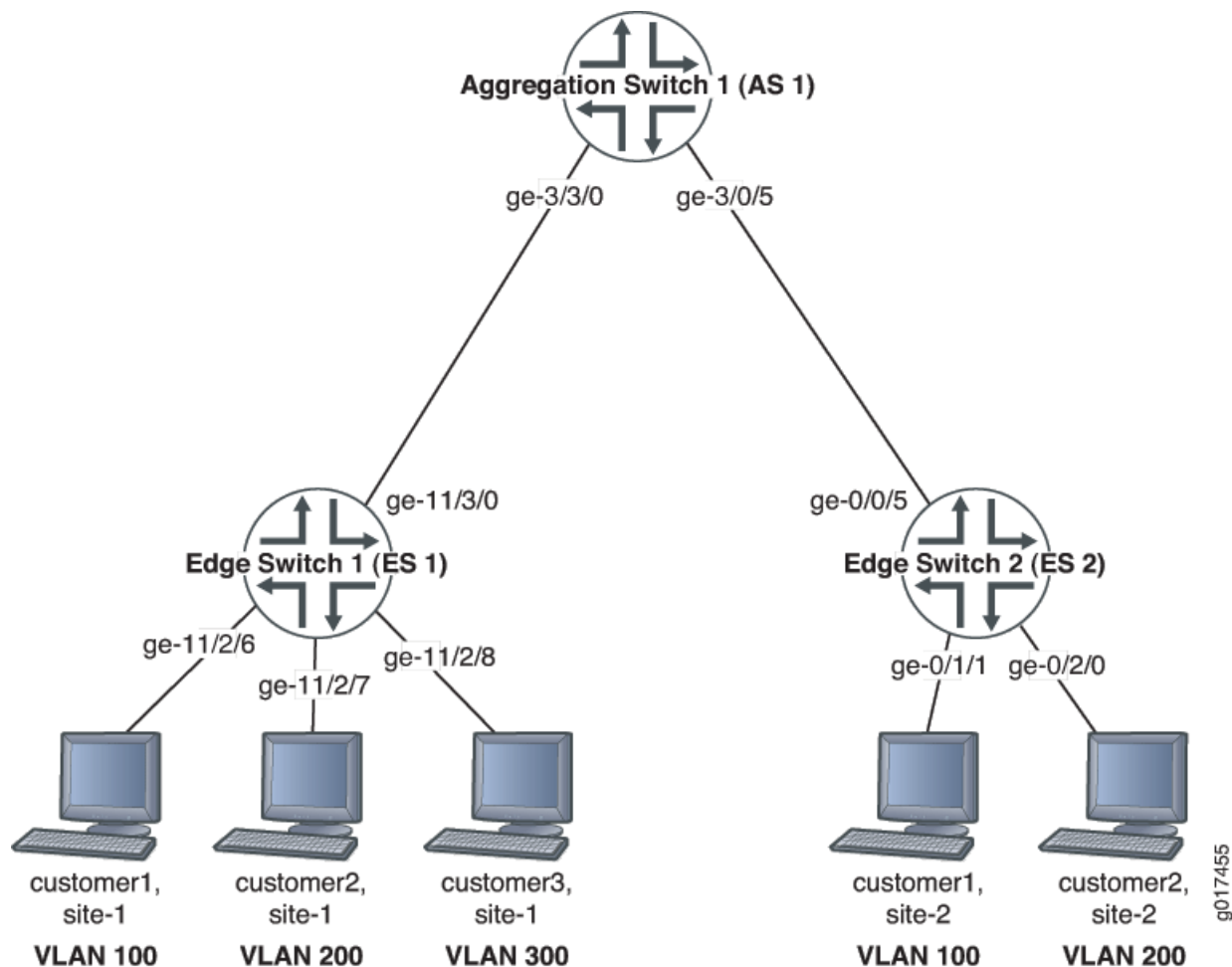


Table 4 on page 52 explains the components of the example topology.

Table 4: Components of the Network Topology

Property	Settings
MX Series routers	<ul style="list-style-type: none"> • ES1 • ES2 • AS1
VLAN tag IDs associated with bridge domain bd	100, 200, and 300

Table 4: Components of the Network Topology (*continued*)

Property	Settings
ES1 interfaces	ES1 interfaces: <ul style="list-style-type: none"> • ge-11/2/6—Access port connecting customer3–site1, VLAN ID 100. • ge-11/2/7—Access port connecting customer2–site1, VLAN ID 200. • ge-11/2/8—Access port connecting customer1–site1, VLAN ID 300. • ge-11/3/0—Trunk port connecting ES1 to AS1.
ES2 interfaces	ES2 interfaces: <ul style="list-style-type: none"> • ge-0/1/1—Access port connecting customer3–site2, VLAN ID 100. • ge-0/2/0—Access port connecting customer3–site2, VLAN ID 200. • ge-0/0/5—Trunk port connecting ES2 to AS1.
AS1 interfaces	AS1 interfaces: <ul style="list-style-type: none"> • ge-3/3/0—Trunk port connected to ES1. • ge-3/0/5—Trunk port connected to ES2.

Configuration

To enable MVRP and RSTP on the trunk interface as well as configure ES1 access interfaces and the bridge domain, perform these tasks:

- Configuring MVRP on ES1 on page 53
- Configuring MVRP on ES2 on page 55
- Configuring MVRP on AS1 on page 57

Configuring MVRP on ES1

CLI Quick Configuration

To quickly configure ES1 for MVRP, copy the following commands and paste them into the switch terminal window of ES1:

```
[edit]
set interfaces ge-11/2/6 description "connected to customer3-site-1"
set interfaces ge-11/2/6 unit 0 family bridge interface-mode access
set interfaces ge-11/2/6 unit 0 family bridge vlan-id 300
set interfaces ge-11/2/7 description "connected to customer2-site-1"
set interfaces ge-11/2/7 unit 0 family bridge interface-mode access
set interfaces ge-11/2/7 unit 0 family bridge vlan-id 200
set interfaces ge-11/2/8 description "connected to customer1-site-1"
set interfaces ge-11/2/8 unit 0 family bridge interface-mode access
set interfaces ge-11/2/8 unit 0 family bridge vlan-id 100
set ge-11/3/0 description "connected to AS1 interface ge-3/3/0"
set ge-11/3/0 unit 0 family bridge interface-mode trunk
set bridge-domains bd vlan-id-list [100 200 300]
set protocols mvrp interface ge-11/3/0
set protocols rstp interface ge-11/3/0
```



NOTE: As we recommend as a best practice, default MVRP timers are used in this example. The default values associated with each MVRP timer are 200 ms for the join timer, 1000 ms for the leave timer, and 10000 ms for the leaveall timer. Modifying timers to inappropriate values might cause an imbalance in the operation of MVRP.

Step-by-Step Procedure

To configure MVRP on ES1:

1. Configure the access interfaces for customers at customer-site 1 and the trunk interface connecting ES1 to AS1:


```
[edit interfaces]
user@es1# set ge-11/2/6 description "connected to customer3-site-1"
user@es1# set ge-11/2/6 unit 0 family bridge interface-mode access
user@es1# set ge-11/2/6 unit 0 family bridge vlan-id 300
user@es1# set ge-11/2/7 description "connected to customer2-site-1"
user@es1# set ge-11/2/7 unit 0 family bridge interface-mode access
user@es1# set ge-11/2/7 unit 0 family bridge vlan-id 200
user@es1# set ge-11/2/8 description "connected to customer1-site-1"
user@es1# set ge-11/2/8 unit 0 family bridge interface-mode access
user@es1# set ge-11/2/8 unit 0 family bridge vlan-id 100
user@es1# set ge-11/3/0 description "connected to AS1 interface ge-3/3/0"
user@es1# set ge-11/3/0 unit 0 family bridge interface-mode trunk
```
2. Configure the bridge domain **bd** and the VLAN IDs associated with the bridge domain:


```
[edit bridge-domains]
user@es1# set bd vlan-id-list [100 200 300]
```
3. Enable MVRP on the trunk interface:


```
[edit protocols]
user@es1# set mvrp interface ge-11/3/0
```
4. Enable RSTP on the trunk interface:


```
[edit protocols]
user@es1# set rstp interface ge-11/3/0
```

Results Check the results of the configuration:

```
user@es1> show configuration
interfaces {
  ge-11/2/6 {
    description "connected to customer3-site-1";
    unit 0 {
      family bridge {
        interface-mode access;
        vlan-id 300;
      }
    }
  }
  ge-11/2/7 {
    description "connected to customer2-site-1";
```

```

    unit 0 {
        family bridge {
            interface-mode access;
            vlan-id 200;
        }
    }
}
ge-11/2/8 {
    description "connected to customer1-site-1";
    unit 0 {
        family bridge {
            interface-mode access;
            vlan-id 100;
        }
    }
}
ge-11/3/0 {
    description "connected to AS1 interface ge-3/3/0";
    unit 0 {
        family bridge {
            interface-mode trunk;
        }
    }
}
}
bridge-domains {
    bd {
        vlan-id-list [ 100 200 300 ];
    }
}
protocols {
    mvrp {
        interface ge-11/3/0;
    }
    rstp {
        interface ge-11/3/0;
    }
}
}

```

Configuring MVRP on ES2

CLI Quick Configuration To quickly configure ES2 for MVRP, copy the following commands and paste them into the switch terminal window of ES2:

```

[edit]
set interfaces ge-0/0/5 description "connected to AS1 interface ge-3/0/5"
set interfaces ge-0/0/5 unit 0 family bridge interface-mode trunk
set interfaces ge-0/1/1 description "connected to customer1-site-2"
set interfaces ge-0/1/1 unit 0 family bridge interface-mode access
set interfaces ge-0/1/1 unit 0 family bridge vlan-id 100
set interfaces ge-0/2/0 description "connected to customer2-site-2"
set interfaces ge-0/2/0 unit 0 family bridge interface-mode access
set interfaces ge-0/2/0 unit 0 family bridge vlan-id 200
set bridge-domains bd vlan-id-list [100 200]
set protocols mvrp interface ge-0/0/5
set protocols rstp interface ge-0/0/5

```



NOTE: As we recommend as a best practice, default MVRP timers are used in this example. The default values associated with each MVRP timer are 200 ms for the join timer, 1000 ms for the leave timer, and 10000 ms for the leaveall timer. Modifying timers to inappropriate values might cause an imbalance in the operation of MVRP.

Step-by-Step Procedure

To enable MVRP and RSTP on the trunk interface as well as configure ES2 access interfaces and the bridge domain, perform these tasks:

1. Configure the access interfaces for customers at customer site **site-2** and the trunk interface connecting ES2 to AS1:

```
[edit interfaces]
user@es2# set ge-0/0/5 description "connected to AS1 interface ge-3/0/5"
user@es2# set ge-0/0/5 unit 0 family bridge interface-mode trunk
user@es2# set ge-0/1/1 description "connected to customer1-site-2"
user@es2# set ge-0/1/1 unit 0 family bridge interface-mode access
user@es2# set ge-0/1/1 unit 0 family bridge vlan-id 100
user@es2# set ge-0/2/0 description "connected to customer2-site-2"
user@es2# set ge-0/2/0 unit 0 family bridge interface-mode access
user@es2# set ge-0/2/0 unit 0 family bridge vlan-id 200
```

2. Configure the bridge domain **bd** and the VLAN IDs associated with the bridge domain:

```
[edit bridge-domains]
user@es2# set bd vlan-id-list [100 200]
```

3. Enable MVRP on the trunk interface:

```
[edit protocols]
user@es2# set mvrp interface ge-0/0/5
```

4. Enable RSTP on the trunk interface:

```
[edit protocols]
user@es2# set rstp interface ge-0/0/5
```

Results Check the results of the configuration:

```
user@es2> show configuration
interfaces {
  ge-0/0/5 {
    description "connected to AS1 interface ge-3/0/5";
    unit 0 {
      family bridge {
        interface-mode trunk;
      }
    }
  }
  ge-0/1/1 {
    description "connected to customer1-site-2";
    unit 0 {
      family bridge {
        interface-mode access;
      }
    }
  }
}
```

```

        vlan-id 100;
    }
}
ge-0/2/0 {
    description "connected to customer2-site-2";
    unit 0 {
        family bridge {
            interface-mode access;
            vlan-id 200;
        }
    }
}
bridge-domains {
    bd {
        vlan-id-list [ 100 200 ];
    }
}
protocols {
    mvrp {
        interface ge-0/0/5;
    }
    rstp {
        interface ge-0/0/5;
    }
}

```

Configuring MVRP on AS1

CLI Quick Configuration

To quickly configure AS1 for MVRP, copy the following commands and paste them into the switch terminal window of AS1:

```

[edit]
set interfaces ge-3/0/5 description "connected to ES2 interface ge-0/0/5"
set interfaces ge-3/0/5 unit 0 family bridge interface-mode trunk
set interfaces ge-3/3/0 description "connected to ES1 interface ge-11/3/0"
set interfaces ge-3/3/0 unit 0 family bridge interface-mode trunk
set protocols mvrp interface ge-3/0/5
set protocols mvrp interface ge-3/3/0
set protocols rstp bridge-priority 0
set protocols rstp interface ge-3/0/5
set protocols rstp interface ge-3/3/0

```



NOTE: As we recommend as a best practice, default MVRP timers are used in this example. The default values associated with each MVRP timer are 200 ms for the join timer, 1000 ms for the leave timer, and 10000 ms for the leaveall timer. Modifying timers to inappropriate values might cause an imbalance in the operation of MVRP.

Step-by-Step Procedure

To enable MVRP and RSTP on the trunk interfaces on AS1, perform these tasks:

1. Configure the trunk interfaces connecting AS1 to ES1 and ES2:

```
[edit interfaces]
user@as1# set ge-3/0/5 description "connected to ES2 interface ge-0/0/5"
user@as1# set ge-3/0/5 unit 0 family bridge interface-mode trunk
user@as1# set ge-3/3/0 description "connected to ES1 interface ge-11/3/0"
user@as1# set ge-3/3/0 unit 0 family bridge interface-mode trunk
```

2. Enable MVRP on the trunk interfaces:

```
[edit protocols]
user@as1# set mvrp interface ge-3/0/5
user@as1# set mvrp interface ge-3/3/0
```

3. Enable RSTP on the trunk interfaces:

```
[edit protocols]
user@as1# set rstp bridge-priority 0
user@as1# set rstp interface ge-3/0/5
user@as1# set rstp interface ge-3/3/0
```

Results Check the results of the configuration:

```
user@as1> show configuration
interfaces {
  ge-3/0/5 {
    description "connected to ES2 interface ge-0/0/5";
    unit 0 {
      family bridge {
        interface-mode trunk;
      }
    }
  }
  ge-3/3/0 {
    description "connected to ES1 interface ge-11/3/0";
    unit 0 {
      family bridge {
        interface-mode trunk;
      }
    }
  }
}
protocols {
  mvrp {
    interface ge-3/0/5;
    interface ge-3/3/0;
  }
  rstp {
    bridge-priority 0;
    interface ge-3/0/5;
    interface ge-3/3/0;
  }
}
```

Verification

To confirm that the configuration is updating VLAN membership, perform these tasks:

- Verifying That MVRP Is Enabled on ES1 on page 59
- Verifying the MVRP Registration on ES1 on page 59
- Verifying Dynamic VLAN Members on ES1 on page 60
- Verifying That MVRP Is Enabled on ES2 on page 60
- Verifying the MVRP Registration on ES2 on page 60
- Verifying Dynamic VLAN Members on ES2 on page 61
- Verifying That MVRP Is Enabled on AS1 on page 61
- Verifying the MVRP Registration on AS1 on page 61
- Verifying That MVRP Is Updating VLAN Membership on AS1 on page 62

Verifying That MVRP Is Enabled on ES1

Purpose Verify that MVRP is enabled on ES1.

Action Show the MVRP applicant state:

```
user@es1> show mvrp applicant-state
MVRP applicant state for routing instance 'default-switch'
(V0) Very anxious observer, (VP) Very anxious passive, (VA) Very anxious new,
(AN) Anxious new, (AA) Anxious active, (QA) Quiet active, (LA) Leaving active,
(AO) Anxious observer, (QO) Quiet observer, (LO) Leaving observer,
(AP) Anxious passive, (QP) Quiet passive
```

VLAN Id	Interface	State
100	ge-11/3/0	Declaring (QA)
200	ge-11/3/0	Declaring (QA)
300	ge-11/3/0	Declaring (QA)

Meaning The output displayed shows that trunk interface **ge-11/3/0** on ES1 is declaring (sending out) interest in the VLAN IDs **100, 200, and 300**.

Verifying the MVRP Registration on ES1

Purpose Verify the VLANs that are registering on ES1.

Action List VLANs in the registered state:

```
user@es1> show mvrp registration-state
MVRP registration state for routing instance 'default-switch'
```

VLAN Id	Interface	Registrar State	Forced State	Managed State	STP State
100	ge-11/3/0	Registered	Registered	Normal	Forwarding
200	ge-11/3/0	Registered	Registered	Normal	Forwarding
300	ge-11/3/0	Empty	Empty	Normal	Forwarding

Meaning The output displayed shows the registrar state for VLANs **100** and **200** is **Registered** indicating that these VLANs are receiving traffic from customer site site-2. VLAN **300** is in an **Empty** state and is not receiving traffic from site-2.

Verifying Dynamic VLAN Members on ES1

Purpose Verify that flooding is not occurring on unregistered VLANs.

Action List dynamic VLAN membership:

```
user@es1> show mvrp dynamic-vlan-memberships
MVRP dynamic vlans for routing instance 'default-switch'
(s) static vlan, (f) fixed registration
```

VLAN Id	Interfaces
100 (s)	ge-11/3/0
200 (s)	ge-11/3/0
300 (s)	

Meaning The output displayed shows that VLAN **300** is not associated with the trunk interface **ge-11/3/0** connected to AS1. No unnecessary traffic is flooding the interface for VLAN **300** towards ES2 site-2.

Verifying That MVRP Is Enabled on ES2

Purpose Verify that MVRP is enabled on ES2.

Action Show the MVRP applicant state:

```
user@es2> show mvrp applicant-state
MVRP applicant state for routing instance 'default-switch'
(V0) Very anxious observer, (VP) Very anxious passive, (VA) Very anxious new,
(AN) Anxious new, (AA) Anxious active, (QA) Quiet active, (LA) Leaving active,
(AO) Anxious observer, (QO) Quiet observer, (LO) Leaving observer,
(AP) Anxious passive, (QP) Quiet passive
```

VLAN Id	Interface	State
100	ge-0/0/5	Declaring (QA)
200	ge-0/0/5	Declaring (QA)
300	ge-0/0/5	Idle (V0)

Meaning The output displayed shows that trunk interface **ge-0/0/5** on ES2 is declaring (sending out) interest in VLAN IDs **100** and **200** but is not declaring interest for VLAN **300**. The state displayed for VLAN **300** is **Idle**.

Verifying the MVRP Registration on ES2

Purpose Verify the VLANs that are registering on ES2.

Action List VLANs in the registered state:

```
user@es2> show mvrp registration-state
MVRP registration state for routing instance 'default-switch'
```

VLAN Id	Interface	Registrar	Forced	Managed	STP
---------	-----------	-----------	--------	---------	-----

		State	State	State	State
100	ge-0/0/5	Registered	Registered	Normal	Forwarding
200	ge-0/0/5	Registered	Registered	Normal	Forwarding
300	ge-0/0/5	Registered	Registered	Normal	Forwarding

Meaning The output displayed shows that the registrar state for VLANs **100**, **200**, and **300** is **Registered** indicating that these VLANs are receiving traffic from customer site site-1.

Verifying Dynamic VLAN Members on ES2

Purpose Verify dynamic VLAN membership.

Action List dynamic VLAN membership:

```
user@es2> show mvrp dynamic-vlan-memberships
MVRP dynamic vlans for routing instance 'default-switch'
(s) static vlan, (f) fixed registration
```

VLAN Id	Interfaces
100 (s)	ge-0/0/5
200 (s)	ge-0/0/5
300	ge-0/0/5

Meaning The output displayed shows that VLAN **300** is not a static VLAN. A static VLAN is (indicated by the **s** beside the VLAN ID. VLAN **300** added to ES2 shows the VLAN membership is being updated.

Verifying That MVRP Is Enabled on AS1

Purpose Verify that MVRP is enabled on AS1.

Action Show the MVRP applicant state:

```
user@es2> show mvrp applicant-state
MVRP applicant state for routing instance 'default-switch'
(V0) Very anxious observer, (VP) Very anxious passive, (VA) Very anxious new,
(AN) Anxious new, (AA) Anxious active, (QA) Quiet active, (LA) Leaving active,
(A0) Anxious observer, (Q0) Quiet observer, (L0) Leaving observer,
(AP) Anxious passive, (QP) Quiet passive
```

VLAN Id	Interface	State
100	ge-3/3/0	Declaring (QA)
	ge-3/0/5	Declaring (QA)
200	ge-3/3/0	Declaring (QA)
	ge-3/0/5	Declaring (QA)
300	ge-3/3/0	Idle (V0)
	ge-3/0/5	Declaring (QA)

Meaning The output displayed shows that trunk interfaces **ge-3/3/0** (connected to ES1) and **ge-3/0/5** (connected to ES2) are declaring (sending out) interest in the VLAN IDs **100** and **200**. Interface **ge-3/0/5** is declaring interest for VLAN **300** (towards ES2) but not declaring interest for VLAN **300** on interface **ge-3/3/0** (towards ES1).

Verifying the MVRP Registration on AS1

Purpose Verify the VLANs that are registering on AS1.

Action List VLANs in the registered state:

```
user@as1> show mvrp registration-state
MVRP registration state for routing instance 'default-switch'
```

VLAN Id	Interface	Registrar State	Forced State	Managed State	STP State
100	ge-3/3/0	Registered	Registered	Normal	Forwarding
	ge-3/0/5	Registered	Registered	Normal	Forwarding
200	ge-3/3/0	Registered	Registered	Normal	Forwarding
	ge-3/0/5	Registered	Registered	Normal	Forwarding
300	ge-3/3/0	Registered	Registered	Normal	Forwarding
	ge-3/0/5	Empty	Empty	Normal	Forwarding

Meaning The output displayed shows that the registrar state for VLANs **100** and **200** is **Registered** on both sides of AS1 (ES1 and ES2) indicating that traffic is being transmitted and received through these VLANs between customer site site-1 and site-2. The registrar state for VLAN **300** is **Registered** on interface **ge-3/3/0** (connected to ES1) but not on interface **ge-3/0/5** (connected to ES2).

Verifying That MVRP Is Updating VLAN Membership on AS1

Purpose Verify that MVRP is updating VLAN membership on AS1 by displaying the dynamic VLAN membership on AS1.

Action List the VLANs on AS1 that were created dynamically using MVRP:

```
user@as1> show mvrp dynamic-vlan-memberships
MVRP dynamic vlans for routing instance 'default-switch'
(s) static vlan, (f) fixed registration
```

VLAN Id	Interfaces
100	ge-3/3/0
	ge-3/0/5
200	ge-3/3/0
	ge-3/0/5
300	ge-3/3/0

Meaning VLANs are only configured statically on the edge switches. The output displayed shows that all VLANs were learned dynamically. No **(s)** is added beside the VLAN IDs, indicating that they were created dynamically and not added statically.

Related Documentation

- Configuring Multiple VLAN Registration Protocol (MVRP) on page 63
- Controlling the Management State of a VLAN in MVRP Configurations on MX Series Routers (CLI Procedure) on page 65
- Verifying That MVRP Is Working Correctly on page 67
- Understanding Multiple VLAN Registration Protocol (MVRP) on MX Series Routers on page 47

Configuring Multiple VLAN Registration Protocol (MVRP)

Multiple VLAN Registration Protocol (MVRP) is used to manage dynamic VLAN registration in Carrier Ethernet network. You can use MVRP on MX Series routers or on EX Series switches.

For information about using MVRP on EX Series switches, see Example: Configuring Automatic VLAN Administration Using MVRP on EX Series Switches.

MVRP is disabled by default on MX Series routers.

To enable MVRP or set MVRP options, follow these instructions:

- Enabling MVRP on page 63
- Disabling MVRP on page 63
- Changing the Registration Mode to Disable Dynamic VLANs on page 63
- Configuring Timer Values on page 64
- Configuring the Multicast MAC address for MVRP on page 64
- Configuring an MVRP Interface as a Point-to-Point Interface on page 64
- Configuring MVRP Tracing Options on page 65

Enabling MVRP

MVRP can only be enabled on trunk interfaces.

To enable MVRP on a specific trunk interface (here, interface **ge-3/0/5**):

```
[edit protocols mvrp]
user@host# set interfaces ge-3/0/5
```

Disabling MVRP

MVRP is disabled by default. You only need to perform this procedure if you have previously enabled MVRP.

To disable MVRP on all trunk interfaces on the router, use one of the following:

```
[edit protocols mvrp]
user@host# deactivate protocols mvrp
user@host# delete protocols mvrp
```

Changing the Registration Mode to Disable Dynamic VLANs

When the registration mode for an interface is set to **normal** (the default), dynamic VLANs are created on interfaces participating in MVRP. The dynamic VLANs created on one router are then propagated by means of MVRP to other routers in a topology.

However, Dynamic VLAN creation through MVRP can be disabled for all trunk interfaces on a router or for individual trunk interfaces.

For information about disabling dynamic VLAN creation on an interface so that the interface does not register and does not participate in MVRP, see “Controlling the

Management State of a VLAN in MVRP Configurations on MX Series Routers (CLI Procedure)” on page 65.

Configuring Timer Values

The timers in MVRP define the amount of time an interface waits to join or leave MVRP or to send or process the MVRP information for the router after receiving an MVRP PDU:

- The join timer controls the amount of time the router waits to accept a registration request.
- The leave timer controls the period of time that the router waits in the Leave state before changing to the unregistered state.
- The leaveall timer controls the frequency with which the LeaveAll messages are communicated.

The default MVRP timer values are 200 ms for the join timer, 1000 ms for the leave timer, and 10000 ms for the leaveall timer.



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

To set the join timer for a specific interface:

```
[edit protocols mvrp]
user@host# set interfaces ge-3/0/5 join-timer 300
```

To set the leave timer for a specific interface:

```
[edit protocols mvrp]
user@host# set interfaces ge-3/0/5 leave-timer 1200
```

To set the leaveall timer for a specific interface:

```
[edit protocols mvrp]
user@host# set interface ge-3/0/5 leaveall-timer 12000
```

Configuring the Multicast MAC address for MVRP

MVRP uses the customer MVRP multicast MAC address when MVRP is enabled. However, you can configure MVRP to instead use the provider MVRP multicast MAC address.

To configure MVRP to use the provider MVRP multicast MAC address:

```
[edit protocols mvrp]
user@host# set bpdu-destination-mac-address provider-bridge-group;
```

Configuring an MVRP Interface as a Point-to-Point Interface

Specify that a configured interface is connected point-to-point. If specified, a point-to-point subset of the MRP state machine provides a simpler and more efficient method to accelerate convergence on the network.

To specify that an MVRP interface is point-to-point (here, interface **ge-3/0/5**):

```
[edit protocols mvrp]
user@host# set interfaces ge-3/0/5 point-to-point;
```

Configuring MVRP Tracing Options

Set MVRP protocol-level tracing options.

To specify MVRP protocol tracing (here, the file is **/var/log/mvrp-log**, size is **2m**, number of files is **28**, the option **world-readable** indicates the log can be read by user, and MVRP is flagging **events**):

```
[edit protocols mvrp]
user@host# edit protocols mvrp traceoptions file /var/log/mvrp-log size 2m files 28
world-readable flag events
```

Related Documentation

- Example: Configuring Automatic VLAN Administration Using MVRP on MX Series Routers on page 50
- Verifying That MVRP Is Working Correctly

Controlling the Management State of a VLAN in MVRP Configurations on MX Series Routers (CLI Procedure)

MX Series routers use Multiple VLAN Registration Protocol (MVRP) to manage dynamic virtual LAN (VLAN) registration in switching networks. Enabling MVRP on trunk interfaces in Carrier Ethernet networks reduces network overhead by limiting the scope of broadcast, unknown unicast, and multicast (BUM) traffic to interested devices only.

Dynamic VLAN registration through MVRP is enabled by default when you enable MVRP on a trunk interface. The trunk interface automatically uses the **normal** registration mode, accepts MVRP messages, and participates in MVRP. The management state in this case is also known as **normal**. However, it can be useful to configure VLAN IDs to bypass the dynamic VLAN registration process for security reasons or when MVRP is not supported on a peer switch. You can change the management state of a VLAN independently to either exclude it entirely from the MVRP registration process and remain in an unregistered state (**forbidden** state) or to force a VLAN to always stay in a registered state and to be declared on all other forwarding ports (**fixed** state).

Three parameters are used to control the management state of a VLAN in an MVRP configuration:

- The VLAN is a member in the interface VLAN ID list (configured at the **[edit interfaces interface-name family bridge vlan-id-list]** hierarchy level).
- The VLAN is a member in the bridge domain VLAN ID list (configured at the **[edit bridge-domain bridge-domain-name vlan-id-list]** hierarchy level).
- The MVRP registration mode is configured for MVRP (configured at the **[edit protocols mvrp interface interface-name registration (normal | restricted | forbidden)]** hierarchy level).

When these three parameters are combined, a VLAN operates with the following MVRP management states:

- If a VLAN ID is present in both the interface and bridge domain VLAN ID list, the VLAN is in a **fixed** management state, irrespective of the MVRP registration mode.
- If a VLAN ID is present in the interface VLAN ID list but not in the bridge domain VLAN ID list and the MVRP registration mode is **forbidden**, the VLAN ID is in a **forbidden** management state. If the MVRP registration mode is not **forbidden**, the VLAN ID is in a **normal** registration state.
- If a VLAN ID is not present in the interface VLAN ID list and the MVRP registration mode is **forbidden** or **restricted**, the VLAN ID is in a **forbidden** management state. Otherwise, it is in a **normal** management state.

Table 1 defines in more detail the MVRP management state for a VLAN when the interface and bridge domain VLAN ID lists and the MVRP registration mode are configured.

Table 5 on page 66 contains the service configured for BEB2 as well as the correlating S-VLAN, I-SID, and B-VLAN.

Table 5: MVRP Management States

VLAN ID present in Interface VLAN ID List?	VLAN ID present in Bridge Domain VLAN ID List?	Interface uses MVRP Normal Registration Mode	Interface uses MVRP Restricted Registration Mode	Interface uses Forbidden Registration Mode
yes	yes	fixed state	fixed state	fixed state
yes	no	normal state	normal state	forbidden state
yes	yes/no	normal state	forbidden state	forbidden state

This topic describes how to configure the management state for VLANs in an MVRP configuration:

- Configure All VLANs to Operate in Normal State on page 66
- Configure VLANs to Operate With Mixed States (Fixed and Normal) on page 67
- Configure VLANs to Operate With Mixed States (Fixed, Normal, and Forbidden) on page 67

Configure All VLANs to Operate in Normal State

- To configure an interface to operate in the normal state, configure the registration state as **normal**:

```
[edit protocols]
user@host# set mvrp interface interface-name registration normal
```

For example, to configure all VLANs on trunk interface **ge-1/0/0** to operate in **normal** state:

```
[edit]
user@host# set interface ge-1/0/0 family bridge trunk
user@host# set protocols mvrp interface ge-1/0/0 registration normal
```

Configure VLANs to Operate With Mixed States (Fixed and Normal)

- To configure an interface to operate in a fixed state, add the VLANs that should operate in a fixed state to the interface VLAN ID list:

```
[edit]
user@host# set interface interface-name family bridge vlan-id-list vlan-ids
user@host# set bridge-domains bridge-domain-name vlan-id-list vlan-ids
```

For example, to configure the first 1024 VLANs on trunk interface **ge-1/0/0.0** to operate in **fixed** state, and the other VLANs to operate in **normal** state:

```
[edit]
user@host# set interface ge-1/0/0.0 family bridge trunk
user@host# set interface ge-1/0/0.0 family bridge vlan-id-list 1-1024
user@host# set bridge-domains bd vlan-id-list 1-1024
user@host# set protocols mvrp interface ge-1/0/0 registration normal
```

Configure VLANs to Operate With Mixed States (Fixed, Normal, and Forbidden)

- To configure an interface to operate in the forbidden state, configure the registration state as **restricted**:

```
[edit protocols]
user@host# set protocols mvrp interface interface-name registration restricted
```

For example, to configure the first 1024 VLANs on trunk interface **ge-1/0/0.0** to operate in **fixed** state, VLAN IDs 1024-2048 to operate in **normal** state, and the remaining VLANs to operate in **forbidden** state:

```
[edit]
user@host# set interface ge-1/0/0.0 family bridge trunk
user@host# set interface ge-1/0/0.0 family bridge vlan-id-list 1-2048
user@host# set bridge-domains bd vlan-id-list 1-1024
user@host# set protocols mvrp interface ge-1/0/0 registration restricted
```

Related Documentation

- Example: Configuring Automatic VLAN Administration Using MVRP on MX Series Routers on page 50
- Configuring Multiple VLAN Registration Protocol (MVRP) on page 63
- Verifying That MVRP Is Working Correctly on page 67
- Understanding Multiple VLAN Registration Protocol (MVRP) on MX Series Routers on page 47

Verifying That MVRP Is Working Correctly

Purpose After configuring your MX Series router to participate in Multiple VLAN Registration Protocol (MVRP), verify that the configuration is properly set and that MVRP messages are being sent and received on your switch.

Action 1. Confirm that the router is declaring Virtual LANs (VLANs).
Show that MVRP is enabled:

```

user@host> show mvrp

MVRP configuration for routing instance 'default-switch'
MVRP dynamic VLAN creation : Enabled
MVRP BPDU MAC address      : Customer bridge group (01-80-C2-00-00-21)
MVRP timers (ms)
  Interface      Join   Leave  LeaveAll
  ge-11/3/0      200   800    10000

```

Show the MVRP applicant state:

```

user@host> show mvrp applicant-state

MVRP applicant state for routing instance 'default-switch'
(VO) Very anxious observer, (VP) Very anxious passive, (VA) Very anxious
new,
(AN) Anxious new, (AA) Anxious active, (QA) Quiet active, (LA) Leaving
active,
(AO) Anxious observer, (QO) Quiet observer, (LO) Leaving observer,
(AP) Anxious passive, (QP) Quiet passive

VLAN Id      Interface      State
  100      ge-11/3/0      Declaring (QA)
  200      ge-11/3/0      Declaring (QA)
  300      ge-11/3/0      Declaring (QA)

```

2. Confirm that VLANs are registered on interfaces.

List VLANs in the registered state:

```

user@host> show mvrp registration-state

MVRP registration state for routing instance 'default-switch'

VLAN Id  Interface  Registrar  Forced  Managed  STP
          State   State      State   State   State
  100    ge-11/3/0  Registered Registered Normal  Forwarding
  200    ge-11/3/0  Registered Registered Normal  Forwarding
  300    ge-11/3/0   Empty      Empty    Normal  Forwarding

```

3. Display a list of VLANs created dynamically.

List dynamic VLAN membership:

```

user@host> show mvrp dynamic-vlan-memberships

MVRP dynamic vlans for routing instance 'default-switch'
(s) static vlan, (f) fixed registration

VLAN Id      Interfaces
  100          ge-3/3/0
                ge-3/0/5
  200          ge-3/3/0
                ge-3/0/5

```

Meaning The output of `show mvrp applicant-state` shows that trunk interface `ge-11/3/0` is declaring (sending out) interest in the VLAN IDs `100`, `200`, and `300` and MVRP is operating properly.

The output of `show mvrp registrant-state` shows the registrar state for VLANs `100` and `200` as **Registered**, indicating that these VLANs are receiving traffic from a customer site. VLAN `300` is in an **Empty** state and is not receiving traffic from a customer site.

The output of the **show mvrp dynamic-vlan-membership** shows that VLANs **100** and **200** are created dynamically (here, on an MX Series router operating as an aggregation switch between MX Series routers operating as edge switches). VLANs created statically are marked with an **(s)** (which is not indicated in this output).

**Related
Documentation**

- [Example: Configuring Automatic VLAN Administration Using MVRP on EX Series Switches](#)
- [Configuring Multiple VLAN Registration Protocol \(MVRP\) \(CLI Procedure\)](#)
- [Controlling the Management State of a VLAN in MVRP Configurations on MX Series Routers \(CLI Procedure\) on page 65](#)

CHAPTER 7

Summary of Multiple VLAN Registration Protocol Configuration Statements

The following sections explain each of the Multiple VLAN Registration Protocol (MVRP) configuration statements. The statements are organized alphabetically.

bpdu-destination-mac-address

Syntax	<code>bpdu-destination-mac-address provider-bridge-group;</code>
Hierarchy Level	[edit logical-systems <i>logical-system-name</i> protocols mvrp], [edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> protocols mvrp] (for virtual switch instance type), [edit protocols mvrp], [edit routing-instances <i>routing-instance-name</i> protocols mvrp] (for virtual switch instance type)
Release Information	Statement introduced in Junos OS Release 10.1 for MX Series routers.
Description	For Multiple VLAN Registration Protocol (MVRP) configurations, specifies the multicast address for MVRP. If configured, the provider MVRP multicast MAC address is used; otherwise, the Junos OS uses the customer MVRP multicast MAC address.
Default	By default, the provider MVRP multicast MAC address is used (if configured). Otherwise, the customer MVRP MAC address is used.
Required Privilege Level	routing—To view this statement in the configuration. routing-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• Example: Configuring Automatic VLAN Administration Using MVRP on MX Series Routers on page 50• Configuring Multiple VLAN Registration Protocol (MVRP) on page 63• Verifying That MVRP Is Working Correctly on page 67• Understanding Multiple VLAN Registration Protocol (MVRP) on MX Series Routers on page 47

interface (MVRP)

Syntax	<pre>interface (all <i>interface-name</i>) { join-timer <i>milliseconds</i>; leave-timer <i>milliseconds</i>; leaveall-timer <i>milliseconds</i>; point-to-point; registration (forbidden normal restricted); }</pre>
Hierarchy Level	<pre>[edit logical-systems <i>logical-system-name</i> protocols mvrp], [edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> protocols mvrp] (for virtual switch instance type), [edit protocols mvrp], [edit routing-instances <i>routing-instance-name</i> protocols mvrp] (for virtual switch instance type)</pre>
Release Information	Statement introduced in Junos OS Release 10.1 for MX Series routers.
Description	Specify interfaces on which to configure Multiple VLAN Registration Protocol (MVRP).
Default	By default, MVRP is disabled.
Options	<p>all—All interfaces on the router.</p> <p><i>interface-name</i>—Names of interface to be configured for MVRP.</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>
Related Documentation	<ul style="list-style-type: none">• Example: Configuring Automatic VLAN Administration Using MVRP on MX Series Routers on page 50• Configuring Multiple VLAN Registration Protocol (MVRP) on page 63• Verifying That MVRP Is Working Correctly on page 67• Understanding Multiple VLAN Registration Protocol (MVRP) on MX Series Routers on page 47

join-timer (MVRP)

Syntax	<code>join-timer <i>milliseconds</i>;</code>
Hierarchy Level	<p>[edit logical-systems <i>logical-system-name</i> protocols mvrp], [edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> protocols mvrp interface (all <i>interface-name</i>)] (for virtual switch instance type), [edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> protocols mvrp] (for virtual switch instance type), [edit logical-systems <i>logical-system-name</i> protocols mvrp interface (all <i>interface-name</i>)], [edit protocols mvrp interface (all <i>interface-name</i>)], [edit routing-instances <i>routing-instance-name</i> protocols mvrp] (for virtual switch instance type), [edit routing-instances <i>routing-instance-name</i> protocols mvrp interface (all <i>interface-name</i>)] (for virtual switch instance type)</p>
Release Information	Statement introduced in Junos OS Release 10.1 for MX Series routers.
Description	For Multiple VLAN Registration Protocol (MVRP), configure the maximum interval interfaces must wait before sending MVRP protocol data units (PDUs).
Default	200 milliseconds
Options	<i>milliseconds</i> —Interval that the interface must wait before sending MVRP PDUs. Maintain default timer settings unless there is a compelling reason to change the settings. Modifying timers to inappropriate values might cause an imbalance in the operation of MVRP.
Required Privilege Level	<p>routing—To view this statement in the configuration. routing-control—To add this statement to the configuration.</p>
Related Documentation	<ul style="list-style-type: none"> • Example: Configuring Automatic VLAN Administration Using MVRP on MX Series Routers on page 50 • Configuring Multiple VLAN Registration Protocol (MVRP) on page 63 • Verifying That MVRP Is Working Correctly on page 67 • Understanding Multiple VLAN Registration Protocol (MVRP) on MX Series Routers on page 47 • leaveall-timer on page 74 • leave-timer on page 75 • point-to-point on page 78 • registration on page 79

leaveall-timer (MVRP)

Syntax	<code>leaveall-timer <i>milliseconds</i>;</code>
Hierarchy Level	<code>[edit logical-systems <i>logical-system-name</i> protocols mvrp],</code> <code>[edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> protocols</code> <code> <i>mvrp</i> interface (all <i>interface-name</i>)] (for virtual switch instance type),</code> <code>[edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> protocols</code> <code> <i>mvrp</i>] (for virtual switch instance type),</code> <code>[edit logical-systems <i>logical-system-name</i> protocols mvrp interface (all <i>interface-name</i>)],</code> <code>[edit protocols mvrp interface (all <i>interface-name</i>)],</code> <code>[edit routing-instances <i>routing-instance-name</i> protocols mvrp] (for virtual switch instance</code> <code> type),</code> <code>[edit routing-instances <i>routing-instance-name</i> protocols mvrp interface (all <i>interface-name</i>)]</code> <code> (for virtual switch instance type)</code>
Release Information	Statement introduced in Junos OS Release 10.1 for MX Series routers.
Description	For Multiple VLAN Registration Protocol (MVRP), configure the interval at which the LeaveAll state operates on the interface.
Default	10000 milliseconds
Options	<i>milliseconds</i> —Interval between the sending of Leave All messages. Maintain default timer settings unless there is a compelling reason to change the settings. Modifying timers to inappropriate values might cause an imbalance in the operation of MVRP.
Required Privilege Level	routing—To view this statement in the configuration. routing-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• Example: Configuring Automatic VLAN Administration Using MVRP on MX Series Routers on page 50• Configuring Multiple VLAN Registration Protocol (MVRP) on page 63• Verifying That MVRP Is Working Correctly on page 67• Understanding Multiple VLAN Registration Protocol (MVRP) on MX Series Routers on page 47• <code>join-timer</code> on page 73• <code>leave-timer</code> on page 75• <code>point-to-point</code> on page 78• <code>registration</code> on page 79

leave-timer (MVRP)

Syntax	<code>leave-timer <i>milliseconds</i>;</code>
Hierarchy Level	<pre>[edit logical-systems <i>logical-system-name</i> protocols mvrp], [edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> protocols mvrp interface (all <i>interface-name</i>)] (for virtual switch instance type), [edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> protocols mvrp] (for virtual switch instance type), [edit logical-systems <i>logical-system-name</i> protocols mvrp interface (all <i>interface-name</i>)], [edit protocols mvrp interface (all <i>interface-name</i>)], [edit routing-instances <i>routing-instance-name</i> protocols mvrp] (for virtual switch instance type), [edit routing-instances <i>routing-instance-name</i> protocols mvrp interface (all <i>interface-name</i>)] (for virtual switch instance type)</pre>
Release Information	Statement introduced in Junos OS Release 10.1 for MX Series routers.
Description	For Multiple VLAN Registration Protocol (MVRP), configure the number of milliseconds the switch retains a VLAN in the Leave state before the VLAN is unregistered. If the interface receives a join message before this timer expires, the VLAN remains registered.
Default	1000 milliseconds
Options	<i>milliseconds</i> —Interval that the switch retains a VLAN in the Leave state before the VLAN is unregistered. At a minimum, set the leave-timer interval at twice the join-timer interval. Maintain default timer settings unless there is a compelling reason to change the settings. Modifying timers to inappropriate values might cause an imbalance in the operation of MVRP.
Required Privilege Level	routing—To view this statement in the configuration. routing-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none"> • Example: Configuring Automatic VLAN Administration Using MVRP on MX Series Routers on page 50 • Configuring Multiple VLAN Registration Protocol (MVRP) on page 63 • Verifying That MVRP Is Working Correctly on page 67 • Understanding Multiple VLAN Registration Protocol (MVRP) on MX Series Routers on page 47 • join-timer on page 73 • leaveall-timer on page 74 • point-to-point on page 78 • registration on page 79

mvrp

Syntax	<pre> mvrp { bpd-destination-mac-address provider-bridge-group; join-timer <i>milliseconds</i>; leave-timer <i>milliseconds</i>; leaveall-timer <i>milliseconds</i>; interface (all <i>interface-name</i>) { join-timer <i>milliseconds</i>; leave-timer <i>milliseconds</i>; leaveall-timer <i>milliseconds</i>; point-to-point; registration (forbidden normal restricted); } no-dynamic-vlan; traceoptions { file <i>filename</i> <files <i>number</i> > <size <i>size</i> > <no-stamp world-readable no-world-readable>; flag <i>flag</i>; } } </pre>
Hierarchy Level	<p>[edit logical-systems <i>logical-system-name</i> protocols], [edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> protocols] (for virtual switch instance type) [edit protocols], [edit routing-instances <i>routing-instance-name</i> protocols] (for virtual switch instance type),</p>
Release Information	Statement introduced in Junos OS Release 10.1 for MX Series routers.
Description	<p>For Layer 2 networks, configure Multiple VLAN Registration Protocol (MVRP) to dynamically share VLAN information and dynamically configure needed VLANs. Maintaining VLAN configurations based on active VLANs reduces the amount of traffic traveling in the network, saving network resources. MVRP is configured on trunk interfaces.</p> <p>The remaining statements are explained separately.</p>
Default	MVRP is disabled by default.
Required Privilege Level	<p>routing—To view this statement in the configuration. routing-control—To add this statement to the configuration.</p>
Related Documentation	<ul style="list-style-type: none"> • Example: Configuring Automatic VLAN Administration Using MVRP on MX Series Routers on page 50 • Configuring Multiple VLAN Registration Protocol (MVRP) on page 63 • Verifying That MVRP Is Working Correctly on page 67 • Understanding Multiple VLAN Registration Protocol (MVRP) on MX Series Routers on page 47

no-dynamic-vlan

Syntax	no-dynamic-vlan;
Hierarchy Level	[edit logical-systems <i>logical-system-name</i> protocols mvrp], [edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> protocols mvrp] (for virtual switch instance type), [edit protocols mvrp], [edit routing-instances <i>routing-instance-name</i> protocols mvrp] (for virtual switch instance type)
Release Information	Statement introduced in Junos OS Release 10.1 for MX Series routers.
Description	<p>Disable the dynamic creation of VLANs using Multiple VLAN Registration Protocol (MVRP) for interfaces participating in MVRP.</p> <p>Dynamic VLAN configuration can be enabled on an interface independent of MVRP. The MVRP dynamic VLAN configuration setting does not override the interface configuration dynamic VLAN configuration setting. If dynamic VLAN creation is disabled on the interface in the interface configuration, no dynamic VLANs are created on the interface, including dynamic VLANs created using MVRP.</p> <p>This option can only be applied globally; it cannot be applied per interface.</p>
Default	If MVRP is enabled, the dynamic creation of VLANs as a result of MVRP protocol exchange messages is enabled.
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"> • Example: Configuring Automatic VLAN Administration Using MVRP on MX Series Routers on page 50 • Configuring Multiple VLAN Registration Protocol (MVRP) on page 63 • Controlling the Management State of a VLAN in MVRP Configurations on MX Series Routers (CLI Procedure) on page 65 • Understanding Multiple VLAN Registration Protocol (MVRP) on MX Series Routers on page 47

point-to-point (MVRP)

Syntax	point-to-point;
Hierarchy Level	[edit logical-systems <i>logical-system-name</i> protocols mvrp interface (all <i>interface-name</i>)], [edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> protocols mvrp interface (all <i>interface-name</i>)] (for virtual switch instance type), [edit protocols mvrp interface (all <i>interface-name</i>)], [edit routing-instances <i>routing-instance-name</i> protocols mvrp interface (all <i>interface-name</i>)] (for virtual switch instance type)
Release Information	Statement introduced in Junos OS Release 10.1 for MX Series routers.
Description	(Optional) For Multiple VLAN Registration Protocol (MVRP) configurations, configure an interface to be recognized as a point-to-point connection. If specified, a point-to-point subset of the MRP state machine is used to provide a simpler and more efficient method to accelerate convergence on the network.
Default	By default, MVRP is disabled.
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: Configuring Automatic VLAN Administration Using MVRP on MX Series Routers on page 50• Configuring Multiple VLAN Registration Protocol (MVRP) on page 63• Verifying That MVRP Is Working Correctly on page 67• Understanding Multiple VLAN Registration Protocol (MVRP) on MX Series Routers on page 47• join-timer on page 73• leaveall-timer on page 74• leave-timer on page 75• registration on page 79

registration

Syntax	registration (forbidden normal restricted);
Hierarchy Level	[edit protocols mvrp interface (all <i>interface-name</i>)]; [edit routing-instances <i>routing-instance-name</i> protocols mvrp interface (all <i>interface-name</i>)] (for virtual switch instance type); [edit logical-systems <i>logical-system-name</i> protocols mvrp interface (all <i>interface-name</i>)], [edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> protocols mvrp interface (all <i>interface-name</i>)] (for virtual switch instance type);
Release Information	Statement introduced in Junos OS Release 10.1 for MX Series routers.
Description	For Multiple VLAN Registration Protocol (MVRP) configurations, configure the registration mode for the interface.
Default	normal
Options	forbidden —The interface or interfaces do not register and do not participate in MVRP. normal —The interface or interfaces accept MVRP messages and participate in MVRP. restricted —The interface or interfaces ignore all MVRP JOIN messages received for VLANs that are not statically configured for MVRP on the interface.
Required Privilege Level	routing—To view this statement in the configuration. routing-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none"> • Example: Configuring Automatic VLAN Administration Using MVRP on MX Series Routers on page 50 • Configuring Multiple VLAN Registration Protocol (MVRP) on page 63 • Verifying That MVRP Is Working Correctly on page 67 • Understanding Multiple VLAN Registration Protocol (MVRP) on MX Series Routers on page 47 • join-timer on page 73 • leaveall-timer on page 74 • leave-timer on page 75 • point-to-point on page 78

traceoptions (MVRP)

Syntax	<pre>traceoptions { file <i>name</i> <size <i>size</i>> <files <i>number</i>> <(world-readable no-world-readable)>; flag <i>flag</i> <flag-modifier> <disable>; }</pre>
Hierarchy Level	[edit logical-systems <i>logical-system-name</i> protocols mvrp], [edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> protocols mvrp] (for virtual switch instance type), [edit protocols mvrp], [edit routing-instances <i>routing-instance-name</i> protocols mvrp] (for virtual switch instance type)
Release Information	Statement introduced in Junos OS Release 10.1 for MX Series routers.
Description	For Multiple VLAN Registration Protocol (MVRP), configure tracing options.
Default	Traceoptions is disabled.
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 within quotation marks. To include the file statement, you must specify a filename. Name of the file to receive the output of the tracing operation. Enclose the name in quotation marks. We recommend that you place MVRP tracing output in the file /var/log/mvrp-log.</p> <p>files <i>number</i>—(Optional) Maximum number of trace files, in the range from 2 through 1000. The default is 1 trace file. When a trace file named trace-file reaches its maximum size, it is renamed trace-file.0, then trace-file.1, and so on, until the maximum number of trace files is reached. Then the oldest trace file is overwritten. If you specify a maximum number of files, you also must specify a maximum file size with the size option.</p> <p>flag <i>flag</i>—Specify which tracing operation to perform. To specify more than one tracing operation, include multiple flag statements. You can include the following flags:</p> <ul style="list-style-type: none">• all—Enable all trace options flags.• error—Trace all failure conditions.• events—Trace process state change and cleanup events.• pdu—Trace RAPS PDU reception and transmission.• socket—Trace socket activity.• state-machine—Trace information about the state machine.• timers—Trace protocol timers.

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. The file size range is from 10240 through 4294967295. The default file size is 1MB.

world-readable—(Optional) Allow any user to read the log file..

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

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

- Related Documentation**
- Example: Configuring Automatic VLAN Administration Using MVRP on MX Series Routers on page 50
 - Configuring Multiple VLAN Registration Protocol (MVRP) on page 63
 - Verifying That MVRP Is Working Correctly on page 67
 - Understanding Multiple VLAN Registration Protocol (MVRP) on MX Series Routers on page 47

PART 3

Layer 2 Port Mirroring

- Layer 2 Port Mirroring Overview on page 85
- Configuring Layer 2 Port Mirroring on page 101
- Examples of Layer 2 Port Mirroring on page 127

CHAPTER 8

Layer 2 Port Mirroring Overview

- Layer 2 Port Mirroring Overview on page 85
- Layer 2 Port Mirroring Properties on page 86
- Layer 2 Port Mirroring Global Instance on page 87
- Layer 2 Port Mirroring Named Instances on page 88
- Layer 2 Port Mirroring Firewall Filters on page 90
- Layer 2 Port Mirroring to Multiple Destinations Using Next-Hop Groups on page 92
- Guidelines for Configuring Layer 2 Port Mirroring on page 93
- Precedence of Multiple Levels of Layer 2 Port Mirroring on a Physical Interface on page 96
- Behavior of Layer 2 Port Mirroring of Logical Interfaces on PE Routers on page 96

Layer 2 Port Mirroring Overview

On routing platforms that contain an Internet Processor II ASIC, you can send a copy of any incoming packet from the routing platform to an external host address or a packet analyzer for analysis. This is known as *port mirroring*. In Junos OS Release 9.3 and later, Juniper Networks MX Series 3D Universal Edge Routers in a Layer 2 environment support port mirroring for Layer 2 bridging traffic and virtual private LAN service (VPLS) traffic. In Junos OS Release 9.4 and later, MX Series routers in a Layer 2 environment also support port mirroring for Layer 2 VPN traffic over a circuit cross-connect (CCC) that transparently connects logical interfaces of the same type.

Layer 2 port mirroring enables you to specify the manner in which incoming and outgoing packets at specified ports are monitored and the manner in which copies of selected packets are forwarded to another destination, where the packets can be analyzed. MX Series routers support Layer 2 port mirroring by performing flow monitoring functions using a class-of-service (CoS) architecture that is in concept similar to, but in particulars different from, other routing platforms.

Like the M120 Multiservice Edge Router and M320 Multiservice Edge Routers, MX Series routers support port mirroring of IPv4, IPv6, and VPLS packets simultaneously. However, the [Junos OS Layer 2 Configuration Guide](#) describes port mirroring only for Layer 2 bridging traffic (**family bridge**), Layer 2 VPLS traffic (**family vpls**) through an MX Series router, and Layer 2 VPN traffic that passes through a CCC (**family ccc**).

For general information about packet flow within MX Series routers and other routers, see the *Junos OS Class of Service Configuration Guide*.

In a Layer 3 environment, MX Series routers support port mirroring of IPv4 (**family inet**) and IPv6 (**family inet6**) traffic. For information about Layer 3 port mirroring, see the *Junos OS Routing Policy Configuration Guide*.

**Related
Documentation**

- Layer 2 Port Mirroring Properties on page 86
- Restrictions on Layer 2 Port Mirroring on page 93
- Application of Layer 2 Port Mirroring Types on page 94

Layer 2 Port Mirroring Properties

Port mirroring specifies the following types of properties:

- Packet-Selection Properties on page 86
- Packet Address Family on page 86
- Mirror Destination Properties on page 87
- Mirror-Once Option on page 87

Packet-Selection Properties

The packet-selection properties of Layer 2 port-mirroring specify how the sampled packets are to be selected for mirroring:

- The number of packets in each sample.
- The number of packets to mirror from each sample.
- The length to which mirrored packets are to be truncated.

Packet Address Family

The packet address family type specifies the type of traffic to be mirrored. In a Layer 2 environment, MX Series routers support port mirroring for the following packet address families:

- Family type **bridge**—For mirroring VPLS traffic when the physical interface is configured with encapsulation type **ethernet-bridge**.
- Family type **ccc**—For mirroring Layer 2 VPN traffic.
- Family type **vpls**—For mirroring VPLS traffic.



NOTE: In typical applications, you send mirrored packets directly to an analyzer or a workstation for analysis, not to another router. If you must send mirrored packets over a network, you should use tunnels. For Layer 2 VPN implementations, you can use the Layer 2 VPN routing instance type **l2vpn** to tunnel the packets to a remote destination.

For information about configuring a routing instance for Layer 2 VPN, see the [Junos OS VPNs Configuration Guide](#). For a detailed Layer 2 VPN example configuration, see the [Junos OS Feature Guides](#). For information about tunnel interfaces, see the [Junos OS Network Interfaces Configuration Guide](#).

Mirror Destination Properties

For a given packet address family, the mirror destination properties of a Layer 2 port-mirroring instance specify how the selected packets are to be sent on a particular physical interface:

- The physical interface on which to send the selected packets.
- Whether filter checking is to be disabled for the mirror destination interface. By default, filter checking is enabled on all



NOTE: If you apply a filter to an interface that is also a Layer 2 port-mirroring destination, a commit failure occurs unless you have disabled filter checking for that mirror destination interface.

Mirror-Once Option

If port mirroring is enabled at both ingress and egress interfaces, you can prevent the MX Series router from sending duplicate packets to the same destination (which would complicate the analysis of the mirrored traffic).



NOTE: The mirror-once port-mirroring option is a global setting. The option is independent of the packet selection properties and the packet family type-specific mirror destination properties.

Related Documentation

- Layer 2 Port Mirroring Overview on page 85
- Restrictions on Layer 2 Port Mirroring on page 93
- Application of Layer 2 Port Mirroring Types on page 94

Layer 2 Port Mirroring Global Instance

On an MX Series router, you can configure a set of port-mirroring properties that implicitly apply to packets received on all ports in the router chassis. This set of port-mirroring properties is the *global instance* of Layer 2 port mirroring for the router.

Within the global instance configuration, you can specify a set of mirror destination properties for each packet address family supported by Layer 2 port mirroring.

For a general description of Layer 2 port-mirroring properties, see “Layer 2 Port Mirroring Properties” on page 86. For a comparison of the types of Layer 2 port mirroring available on an MX Series router, see “Application of Layer 2 Port Mirroring Types” on page 94.

Related Documentation

- Layer 2 Port Mirroring Overview on page 85
- Configuring the Global Instance of Layer 2 Port Mirroring on page 101
- Examples: Layer 2 Port-Mirroring at Multiple Levels of the Chassis on page 127
- Example: Layer 2 Port Mirroring with Multiple Instances on page 129
- Example: Layer 2 Port Mirroring to Multiple Destinations on page 140

Layer 2 Port Mirroring Named Instances

This topic describes the following information:

- Layer 2 Port Mirroring Named Instances Overview on page 88
- Mirroring at Ports Grouped at the FPC Level on page 89
- Mirroring at Ports Grouped at the PIC Level on page 89
- Mirroring at a Group of Ports Bound to Multiple Named Instances on page 89

Layer 2 Port Mirroring Named Instances Overview

On an MX Series router, you can define a set of port-mirroring properties that you can explicitly bind to physical ports on the router. This set of port mirroring properties is known as a *named instance* of Layer 2 port mirroring.

You can bind a named instance of Layer 2 port mirroring to physical ports associated with MX Series router Packet Forwarding Engine components at different levels of the router chassis:

- At the FPC level—You can bind a named instance to the physical ports associated with a specific Dense Port Concentrator (DPC) or to the physical ports associated with a specific Flexible Port Concentrator (FPC).
- At the PIC level—You can bind a named instance of port mirroring to a specific Packet Forwarding Engine (on a specific DPC) or to a specific PIC.



NOTE: MX Series routers support DPCs as well as FPCs and PICs. Unlike FPCs, DPCs do not support PICs. In the Junos OS CLI, however, you use FPC and PIC syntax to configure or display information about DPCs and the Packet Forwarding Engines on the DPCs.

The following points summarize the behavior of Layer 2 port mirroring based on named instances:

- The scope of packet selection is determined by the target of the binding—At the ports (or port) bound to a named instance of Layer 2 port mirroring, the router selects input packets according to the packet-selection properties in the named instance.
- The destination of a selected packet is determined by the packet address family—Of the packets selected, the router mirrors only the packets belonging to an address family

for which the named instance of Layer 2 port mirroring specifies a set of mirror destination properties. In a Layer 2 environment, MX Series routers support port mirroring of VPLS (**family bridge** or **family vpls**) traffic and Layer 2 VPN traffic with **family ccc**.

For a general description of Layer 2 port-mirroring properties, see “Layer 2 Port Mirroring Properties” on page 86. For a comparison of the types of Layer 2 port mirroring available on an MX Series router, see “Application of Layer 2 Port Mirroring Types” on page 94.

Mirroring at Ports Grouped at the FPC Level

On an MX Series router, you can bind a named instance of Layer 2 port mirroring to a specific DPC or FPC installed in the router chassis. The port mirroring properties in the instance are applied to all Packet Forwarding Engines (and their associated ports) on the specified DPC or to all PICs (and their associated ports) installed in the specified FPC. Port mirroring properties that are bound to a DPC or FPC override any port-mirroring properties bound at the global level or the MX Series router chassis.

Mirroring at Ports Grouped at the PIC Level

On an MX Series router, you can bind a named instance of Layer 2 port mirroring to a specific Packet Forwarding Engine or PIC. The port-mirroring properties in that instance are applied to all ports associated with the specified Packet Forwarding Engine or PIC. Port-mirroring properties that are bound to a Packet Forwarding Engine or PIC override any port-mirroring properties bound at the DPC or FPC that contains them.



NOTE: For MX960 routers, there is a one-to-one mapping of Packet Forwarding Engines to Ethernet ports. Therefore, on MX960 routers only, you can configure port-specific bindings of port-mirroring instances.

Mirroring at a Group of Ports Bound to Multiple Named Instances

On an MX Series router, you can apply up to two named instances of Layer 2 port mirroring to the same group of ports within the router chassis. By applying two different port-mirroring instances to the same DPC, FPC, Packet Forwarding Engine, or PIC, you can bind two distinct Layer 2 port mirroring specifications to a single group of ports.



NOTE: You can configure only one global instance of Layer 2 port mirroring on an MX Series router.

Related Documentation

- Layer 2 Port Mirroring Overview on page 85
- Defining a Named Instance of Layer 2 Port Mirroring on page 104
- Binding Layer 2 Port Mirroring to Ports Grouped at the FPC Level on page 108
- Binding Layer 2 Port Mirroring to Ports Grouped at the PIC Level on page 110
- Examples: Layer 2 Port-Mirroring at Multiple Levels of the Chassis on page 127
- Example: Layer 2 Port Mirroring with Multiple Instances on page 129

Layer 2 Port Mirroring Firewall Filters

This topic describes the following information:

- Layer 2 Port Mirroring Firewall Filters Overview on page 90
- Mirroring of Packets Received or Sent on a Logical Interface on page 91
- Mirroring of Packets Forwarded or Flooded to a Bridge Domain on page 91
- Mirroring of Packets Forwarded or Flooded to a VPLS Routing Instance on page 91

Layer 2 Port Mirroring Firewall Filters Overview

On an MX Series router, you can configure a firewall filter *term* to specify that Layer 2 port mirroring is to be applied to all packets at the interface to which the firewall filter is applied.

You can apply a Layer 2 port-mirroring firewall filter to the input or output logical interfaces (including aggregated Ethernet logical interfaces), to traffic forwarded or flooded to a bridge domain, or traffic forwarded or flooded to a VPLS routing instance.

MX Series routers support Layer 2 port mirroring of VPLS (**family bridge** or **family vpls**) traffic and Layer 2 VPN traffic with **family ccc n** in a Layer 2 environment.

Within a firewall filter **term**, you can specify the Layer 2 port-mirroring properties under the **then** statement in either of the following ways:

- Implicitly reference the Layer 2 port mirroring properties in effect on the port.
- Explicitly reference a particular named instance of Layer 2 port mirroring.



NOTE: When configuring a Layer 2 port-mirroring firewall filter, do not include the optional **from** statement that specifies match conditions based on the route source address. Omit this statement so that all packets are considered to match and all *actions* and *action-modifiers* specified in the **then** statement are taken.

If you want to mirror all incoming packets, then you must not use the **from** statement; /* comment: one configure filter terms with **from** if they are interested in mirroring only a subset of packet.

For a general description of Layer 2 port-mirroring properties, see “Layer 2 Port Mirroring Properties” on page 86. For a comparison of the types of Layer 2 port mirroring available on an MX Series router, see “Application of Layer 2 Port Mirroring Types” on page 94.



NOTE: If you associate integrated routing and bridging (IRB) with the bridge domain (or VPLS routing instance), and also configure within the bridge domain (or VPLS routing instance) a forwarding table filter with the `port-mirror` or `port-mirror-instance` action, then the IRB packet is mirrored as a Layer 2 packet. You can disable this behavior by configuring the `no-irb-layer-2-copy` statement in the bridge-domain (or VPLS routing instance).

For a detailed description of how to configure a Layer 2 port-mirroring firewall filter, see “Defining a Layer 2 Port-Mirroring Firewall Filter” on page 113.

For detailed information about how you can use Layer 2 port-mirroring firewall filters with MX Routers configured as provider edge (PE) routers, see “Layer 2 Port Mirroring of PE Router Logical Interfaces” on page 96. For detailed information about configuring firewall filters in general (including in a Layer 3 environment), see the *Junos OS Routing Policy Configuration Guide*.

Mirroring of Packets Received or Sent on a Logical Interface

To mirror Layer 2 traffic received or sent on a logical interface, apply a port-mirroring firewall filter to the input or output of the interface.

A port-mirroring firewall filter can also be applied to an aggregated-Ethernet logical interface. For details, see “Layer 2 Port Mirroring of PE Router Aggregated Ethernet Interfaces” on page 98.



NOTE: If port-mirroring firewall filters are applied at both the input and output of a logical interface, two copies of each packet are mirrored. To prevent the router from forwarding duplicate packets to the same destination, you can enable the “mirror-once” option for Layer 2 port mirroring in the global instance for the Layer 2 packet address family.

Mirroring of Packets Forwarded or Flooded to a Bridge Domain

To mirror Layer 2 traffic forwarded to or flooded to a bridge domain, apply a port-mirroring firewall filter to the input to the forwarding table or flood table. Any packet received for the bridge domain forwarding or flood table and that matches the filter conditions is mirrored.

For more information about bridge domains, see “Layer 2 Bridge Domains Overview” on page 147. For information about flooding behavior in a bridge domain, see “Layer 2 Learning and Forwarding for Bridge Domains Overview” on page 149.

Mirroring of Packets Forwarded or Flooded to a VPLS Routing Instance

To mirror Layer 2 traffic forwarded to or flooded to a VPLS routing instance, apply a port-mirroring firewall filter to the input to the forwarding table or flood table. Any packet received for the VPLS routing instance forwarding or flood table and that matches the filter condition is mirrored.

For more information about VPLS routing instances, see “Configuring a VPLS Routing Instance” on page 16 and “Configuring VLAN Identifiers for Bridge Domains and VPLS Routing Instances” on page 152. For information about flooding behavior in VPLS, see the [Junos OS VPNs Configuration Guide](#).

**Related
Documentation**

- Layer 2 Port Mirroring Overview on page 85
- Defining a Layer 2 Port-Mirroring Firewall Filter on page 113
- Example: Layer 2 Port Mirroring at a Logical Interface on page 133
- Example: Layer 2 Port Mirroring for a Layer 2 VPN on page 135
- Example: Layer 2 Port Mirroring for a Layer 2 VPN with LAG Links on page 138
- Example: Layer 2 Port Mirroring to Multiple Destinations on page 140

Layer 2 Port Mirroring to Multiple Destinations Using Next-Hop Groups

On an MX Series router, you can mirror traffic to multiple destinations by configuring next-hop groups in Layer 2 port-mirroring firewall filters applied to tunnel interfaces. The mirroring of packets to multiple destinations is also known as *multipacket port mirroring*.



NOTE: Junos OS Release 9.5 introduced support for Layer 2 port mirroring using next-hop groups on MX Series routers, but required installation of a Tunnel PIC. Beginning in Junos OS Release 9.6, Layer 2 port mirroring using next-hop groups on MX Series routers does not require Tunnel PICs.

On MX Series routers, you can define a firewall filter for mirroring packets to a next-hop group. The next-hop group can contain Layer 2 members, Layer 3 members, and subgroups that are either unit list (mirroring packets to each interface) or load-balanced (mirroring packets to one of several interfaces). The MX Series router supports up to 30 next-hop groups. Each next-hop group supports up to 16 next-hop addresses. Each next-hop group must specify at least two addresses.

To enable port mirroring to the members of a next-hop group, you specify the next-hop group as the filter action of a firewall filter, and then you apply the firewall filter to logical tunnel interfaces (**lt-**) or virtual tunnel interfaces (**vt-**) on the MX Series router.



NOTE: The use of subgroups for load-balancing mirrored traffic is not supported.

**Related
Documentation**

- Layer 2 Port Mirroring Overview on page 85
- Defining a Layer 2 Port-Mirroring Firewall Filter on page 113
- Defining a Next-Hop Group for Layer 2 Port Mirroring on page 124
- Example: Layer 2 Port Mirroring to Multiple Destinations on page 140

Guidelines for Configuring Layer 2 Port Mirroring

The following topics describe general guidelines for configuring Layer 2 port mirroring on an MX Series router:

- Restrictions on Layer 2 Port Mirroring on page 93
- Application of Layer 2 Port Mirroring Types on page 94

Restrictions on Layer 2 Port Mirroring

The following restrictions apply to Layer 2 port mirroring:

- Only Layer 2 transit data (packets that contain chunks of data transiting the routing platform as they are forwarded from a source to a destination) can be mirrored. Layer 2 local data (packets that contain chunks of data that are destined for or sent by the Routing Engine, such as Layer 2 control packets) are not mirrored.
- If you apply a port-mirroring filter to the output of a logical interface, only unicast packets are mirrored. To mirror broadcast packets, multicast packets, unicast packets with an unknown destination media access control (MAC) address, or packets with MAC entry in the destination MAC (DMAC) routing table, apply a filter to the input to the flood table of a bridge domain or virtual private LAN service (VPLS) routing instance.
- The mirror destination device should be on a dedicated bridge domain and should not participate in any bridging activity: The mirror destination device should not have a bridge to the ultimate traffic destination, and the mirror destination device should not send the mirrored packets back to the source address.
- For either the global port-mirroring instance or a named port-mirroring instance, you can configure only one mirror output interface per port-mirroring instance and packet address family. If you include more than one **interface** statement under the **family (bridge | ccc | vpls) output** statement, the previous **interface** statement is overridden.
- Layer 2 port-mirroring firewall filtering is not supported for logical systems.

In a Layer 2 port-mirroring firewall filter definition, the filter **action-modifier** (**port-mirror** or **port-mirror-instance pm-instance-name**) relies on port-mirroring properties defined in the global instance or named instances of Layer 2 port mirroring, which are configured under the **[edit forwarding-options port-mirroring]** hierarchy. Therefore, the filter **term** cannot support Layer 2 port mirroring for logical systems.

- For a Layer 2 port mirroring firewall filter in which you implicitly reference Layer 2 port mirroring properties by including the **port-mirror** statement, if multiple named instances of Layer 2 port mirroring are bound to the underlying physical interface, then only the first binding in the stanza (or the only binding) is used at the logical interface. This is done mainly for backward compatibility.
- Layer 2 port-mirroring firewall filters do not support the use of next-hop subgroups for load-balancing mirrored traffic.

- Related Documentation**
- Layer 2 Port Mirroring Overview on page 85
 - Application of Layer 2 Port Mirroring Types on page 94
 - Precedence of Multiple Levels of Layer 2 Port Mirroring on a Physical Interface on page 96
 - Layer 2 Port Mirroring of PE Router Logical Interfaces on page 96
 - Layer 2 Port Mirroring of PE Router Aggregated Ethernet Interfaces on page 98

Application of Layer 2 Port Mirroring Types

You can apply different sets of Layer 2 port-mirroring properties to the VPLS packets at different ingress or egress points of an MX Series route.

Table 6 on page 94 describes the three types of Layer 2 port mirroring you can configure on an MX Series router: the global instance, named instances, and firewall filters.

Table 6: Application of Layer 2 Port Mirroring Types

Type of Layer2PortMirroring Definition	Point of Application	Scope of Mirroring	Description	Configuration Details
Global Instance of Layer2PortMirroring	All ports in the MX Series router chassis	VPLS packets received on all ports in the MX Series router chassis	If configured, the global port-mirroring properties implicitly apply to all VPLS packets received on all ports in the router chassis.	See “Configuring the Global Instance of Layer 2 Port Mirroring” on page 101
Named Instance of Layer2PortMirroring	Ports grouped at the FPC level See “Binding Layer 2 Port Mirroring to Ports Grouped at the FPC Level” on page 108.	VPLS packets received on ports associated with a specific DPC or FPC and its Packet Forwarding Engines.	Overrides any port-mirroring properties configured by the global port-mirroring instance.	See “Defining a Named Instance of Layer 2 Port Mirroring” on page 104. NOTE: The number of port-mirroring destinations supported for an MX Series router is limited to the number of Packet Forwarding Engines contained on the DPCs installed in the router chassis.
	Ports grouped at the PIC level See “Binding Layer 2 Port Mirroring to Ports Grouped at the PIC Level” on page 110.	VPLS packets received on ports associated with a specific Packet Forwarding Engine.	Overrides any port-mirroring properties configured at the FPC level or in the global port-mirroring instance.	

Table 6: Application of Layer 2 Port Mirroring Types (*continued*)

Type of Layer2PortMirroring Definition	Point of Application	Scope of Mirroring	Description	Configuration Details
Layer2PortMirroring Firewall Filter	Logical interface (including an aggregated Ethernet interface)	VPLS packets received or sent on a logical interface.	In the firewall filter configuration, include <i>action</i> and <i>action-modifier</i> terms to apply to the packets selected for mirroring:	See “Defining a Layer 2 Port-Mirroring Firewall Filter” on page 113.
	See “Applying Layer 2 Port Mirroring to a Logical Interface” on page 117.		<ol style="list-style-type: none"> 1. The accept action is recommended. 2. Specify port mirroring by Including one of the following modifiers: <ul style="list-style-type: none"> • The port-mirror modifier implicitly references the port-mirroring properties currently bound to the underlying physical interfaces. • The port-mirror-instance <i>pm-instance-name</i> modifier explicitly references a named instance of port mirroring. 3. (Optional) For tunnel interface input packets only, to mirror the packets to additional destinations, include the next-hop-group <i>next-hop-group-name</i> modifier. This modifier references a next-hop-group that specifies the next-hop addresses (for sending additional copies of packets to an analyzer). 	<p>NOTE: Layer 2 port-mirroring firewall filters are not supported for logical systems.</p> <p>For mirroring tunnel interface input packets to multiple destinations, also see “Defining a Next-Hop Group for Layer 2 Port Mirroring” on page 124.</p>
	Bridge domain forwarding table or flood table	Layer 2 traffic forwarded or flooded to a bridge domain		
	See “Applying Layer 2 Port Mirroring to Traffic Forwarded or Flooded to a Bridge Domain” on page 120.			
	VPLS routing instance forwarding table or flood table	Layer 2 traffic forwarded or flooded to a VPLS routing instance		
	See “Applying Layer 2 Port Mirroring to Traffic Forwarded or Flooded to a VPLS Routing Instance” on page 122.			

Related Documentation

- Layer 2 Port Mirroring Overview on page 85
- Restrictions on Layer 2 Port Mirroring on page 93
- Precedence of Multiple Levels of Layer 2 Port Mirroring on a Physical Interface on page 96
- Layer 2 Port Mirroring of PE Router Logical Interfaces on page 96
- Layer 2 Port Mirroring of PE Router Aggregated Ethernet Interfaces on page 98

Precedence of Multiple Levels of Layer 2 Port Mirroring on a Physical Interface

You can bind different sets of Layer 2 port mirroring properties (the global instance and one or more named instances) at various levels of an MX Series router chassis (at the chassis level, at the FPC level, or at the PIC level). Therefore, it is possible for a single group of physical interfaces to be bound to multiple Layer 2 port mirroring definitions.

If a group of ports (or, in the case of a PIC-level binding in an MX960 router, a single port) is bound to multiple Layer 2 port mirroring definitions, the router applies the Layer 2 port-mirroring properties to those ports as follows:

1. **Chassis-level port-mirroring properties implicitly apply to all ports in the chassis.** If an MX Series router is configured with the global port-mirroring instance, those port mirroring properties apply to all ports. See “Configuring the Global Instance of Layer 2 Port Mirroring” on page 101.
2. **FPC-level port-mirroring properties override chassis-level properties.** If a DPC or FPC is bound to a named instance of port mirroring, those port mirroring properties apply to all ports associated with that DPC or FPC, overriding any port mirroring properties bound at the chassis level. See “Binding Layer 2 Port Mirroring to Ports Grouped at the FPC Level” on page 108.
3. **PIC-level port-mirroring properties override FPC-level properties.** If a Packet Forwarding Engine or PIC is bound to a named instance of port mirroring, those port mirroring properties apply to all ports associated with the Packet Forwarding Engine or PIC, overriding any port mirroring properties bound to those ports at the FPC level. See “Binding Layer 2 Port Mirroring to Ports Grouped at the PIC Level” on page 110.

Related Documentation

- Layer 2 Port Mirroring Overview on page 85
- Restrictions on Layer 2 Port Mirroring on page 93
- Application of Layer 2 Port Mirroring Types on page 94
- Layer 2 Port Mirroring of PE Router Logical Interfaces on page 96
- Layer 2 Port Mirroring of PE Router Aggregated Ethernet Interfaces on page 98

Behavior of Layer 2 Port Mirroring of Logical Interfaces on PE Routers

The following topics describe the behavior of Layer 2 port-mirroring firewall filtering on MX Series routers configured as provider edge (PE) routers connected to customer edge (CE) devices, such as routers and Ethernet switches.

- Layer 2 Port Mirroring of PE Router Logical Interfaces on page 96
- Layer 2 Port Mirroring of PE Router Aggregated Ethernet Interfaces on page 98

Layer 2 Port Mirroring of PE Router Logical Interfaces

For an MX Series router configured as a provider edge (PE) router on the customer-facing edge of a service provider network, you can apply a Layer 2 port-mirroring firewall filter

at the following ingress and egress points to mirror the traffic between the MX Series router and customer edge (CE) devices, such as routers and Ethernet switches.

Table 7 on page 97 describes the ways in which you can apply Layer 2 port-mirroring firewall filters to an MX Series router configured as a PE router.

Table 7: Application of Layer 2 Port Mirroring Firewall Filters on PE Routers

Point of Application	Scope of Mirroring	Notes	Configuration Details
Ingress Customer-Facing Logical Interface	Packets originating within a service provider customer's network, sent first to a CE device, and sent next to the MX Series PE router.	<p>You can also configure aggregated Ethernet interfaces between CE devices and PE routers for VPLS routing instances. Traffic is load-balanced across all of the links in the aggregated interface.</p> <p>Traffic received on an aggregated Ethernet interface is forwarded over a different interface based on a lookup of the destination MAC (DMAC) address:</p> <ul style="list-style-type: none"> • Packets destined for a local site are sent out of the load-balanced child interface. • Packets destined for the remote site are encapsulated and forwarded over a label-switched path (LSP). 	<p>See "Applying Layer 2 Port Mirroring to a Logical Interface" on page 117.</p> <p>For more information about VPLS routing instances, see "Configuring a VPLS Routing Instance" on page 16 and "Configuring VLAN Identifiers for Bridge Domains and VPLS Routing Instances" on page 152.</p>
Egress Customer-Facing Logical Interface	<p>Unicast packets being forwarded by the MX Series router to another PE router.</p> <p>NOTE: If you apply a port-mirroring filter to the output for a logical interface, only unicast packets are mirrored. To mirror multicast, unknown unicast, and broadcast packets, apply a filter to the input to the flood table of a bridge domain or VPLS routing instance.</p>		See "Applying Layer 2 Port Mirroring to a Logical Interface" on page 117.
Input to a Bridge Domain Forwarding Table or Flood Table	Forwarding traffic or flood traffic sent to the bridge domain from a CE device.	Forwarding and flood traffic typically consists of broadcast packets, multicast packets, unicast packets with an unknown destination MAC address, or packets with a MAC entry in the DMAC routing table.	See "Applying Layer 2 Port Mirroring to Traffic Forwarded or Flooded to a Bridge Domain" on page 120. For information about flooding behavior in VPLS, see the Junos OS VPNs Configuration Guide .
Input to a VPLS Routing Instance Forwarding Table or Flood Table	Forwarding traffic or flood traffic sent to the VPLS routing instance from a CE device.		See "Applying Layer 2 Port Mirroring to Traffic Forwarded or Flooded to a VPLS Routing Instance" on page 122. For information about flooding behavior in VPLS, see the Junos OS VPNs Configuration Guide .

Related Documentation

- Layer 2 Port Mirroring Overview on page 85
- Layer 2 Port Mirroring Firewall Filters on page 90
- Defining a Layer 2 Port-Mirroring Firewall Filter on page 113
- Example: Layer 2 Port Mirroring at a Logical Interface on page 133

Layer 2 Port Mirroring of PE Router Aggregated Ethernet Interfaces

An aggregated Ethernet interface is a virtual aggregated link that consists of a set of physical interfaces of the same speed and operating in full-duplex link connection mode. You can configure aggregated Ethernet interfaces between CE devices and PE routers for VPLS routing instances. Traffic is load-balanced across all of the links in the aggregated interface. If one or more links in the aggregated interface fails, the traffic is switched to the remaining links.

You can apply a Layer 2 port-mirroring firewall filter to an aggregated Ethernet interface to configure port-mirroring at the parent interface. However, if any child interfaces are bound to different Layer 2 port-mirroring instances, packets received at the child interfaces will be mirrored to the destinations specified by their respective port-mirroring instances. Thus, multiple child interfaces can mirror packets to multiple destinations.

For example, suppose the parent aggregated Ethernet interface instance **ae0** has two child interfaces:

- **xe-2/0/0**
- **xe-3/1/2**

Suppose that these child interfaces on **ae0** are bound to two different Layer 2 port-mirroring instances:

- **pm_instance_A**—A named instance of Layer 2 port-mirroring, bound to child interface **xe-2/0/0**.
- **pm_instance_B**—A named instance of Layer 2 port-mirroring, bound to child interface **xe-3/1/2**.

Now suppose you apply a Layer 2 port-mirroring firewall filter to the Layer 2 traffic sent on **ae0.0** (logical unit **0** on the aggregated Ethernet interface instance **0**). This enables port mirroring on **ae0.0**, which has the following effect on the processing of traffic received on the child interfaces for which Layer 2 port-mirroring properties are specified:

- The packets received on **xe-2/0/0.0** are mirrored to the output interfaces configured in port-mirroring instance **pm_instance_A**.
- The packets received on **xe-3/1/2.0** are mirrored to the output interfaces configured in port-mirroring instance **pm_instance_B**.

Because **pm_instance_A** and **pm_instance_B** can specify different packet-selection properties or mirror destination properties, the packets received on **xe-2/0/0.0** and **xe-3/1/2.0** can mirror different packets to different destinations.

**Related
Documentation**

- Layer 2 Port Mirroring Overview on page 85
- Layer 2 Port Mirroring Firewall Filters on page 90
- Defining a Layer 2 Port-Mirroring Firewall Filter on page 113
- Example: Layer 2 Port Mirroring for a Layer 2 VPN on page 135

CHAPTER 9

Configuring Layer 2 Port Mirroring

- Configuring Layer 2 Port Mirroring for Physical Interfaces on page 101
- Configuring Layer 2 Port Mirroring for Logical Interfaces on page 113
- Configuring Layer 2 Port Mirroring to Multiple Destinations on page 124

Configuring Layer 2 Port Mirroring for Physical Interfaces

The following topics describe the configuration of Layer 2 port mirroring for physical interfaces in MX Series routers:

- Configuring the Global Instance of Layer 2 Port Mirroring on page 101
- Defining a Named Instance of Layer 2 Port Mirroring on page 104
- Displaying Information About DPCs or FPCs in an MX Series Router on page 107
- Binding Layer 2 Port Mirroring to Ports Grouped at the FPC Level on page 108
- Binding Layer 2 Port Mirroring to Ports Grouped at the PIC Level on page 110
- Displaying Layer 2 Port-Mirroring Instance Settings and Status on page 111
- Disabling Layer 2 Port Mirroring Instances on page 112

Configuring the Global Instance of Layer 2 Port Mirroring

On an MX Series router, you can configure a set of Layer 2 port-mirroring properties that implicitly apply to packets received on all ports in the router chassis.

To configure the global instance of Layer 2 port mirroring on an MX Series router:

1. Enable configuration of the Layer 2 port mirroring:

```
[edit]  
user@host# edit forwarding-options port-mirroring
```
2. Enable configuration of the packet-selection properties:

```
[edit forwarding-options port-mirroring]  
user@host# edit input
```

3. Specify global-level packet-selection properties.

a. Specify the number of packets to select:

```
[edit forwarding-options port-mirroring input]  
user@host# set rate number
```

The valid range is 1 through 65535.

b. Specify the number of packets to mirror from each selection:

```
[edit forwarding-options port-mirroring input]  
user@host# set run-length number
```

The valid range is 0 through 20. The default value is 0.

c. Specify the length to which mirrored packets are to be truncated:

```
[edit forwarding-options port-mirroring input]  
user@host# set maximum-packet-length number
```

The valid range is 0 through 9216. The default value is 0, which means the mirrored packets are not truncated.

4. Specify the global-level Layer 2 address-type family from which traffic is to be selected for mirroring:

```
[edit forwarding-options port-mirroring input]  
user@host# up  
[edit forwarding-options port-mirroring]  
user@host# edit family family
```

The value of the *family* option can be **bridge**, **ccc**, or **vpls**.



NOTE: Under the [edit forwarding-options port-mirroring] hierarchy level, the protocol family statement **family bridge** is an alias for **family vpls**. The command-line interface (CLI) displays Layer 2 port-mirroring configurations as **family vpls**, even for Layer 2 port-mirroring configured as **family bridge**. Use **family bridge** when the physical interface is configured with **encapsulation ethernet-bridge**.

5. Enable configuration of global-level mirror destination properties for this address family:

```
[edit forwarding-options port-mirroring family family]  
user@host# edit output
```

6. Specify global-level mirror destination properties for this address family.

- a. Specify the physical interface on which to send the mirrored packets:

```
[edit forwarding-options port-mirroring family family output]
user@host# set interface interface-name
```

You can also specify an integrated routing and bridging (IRB) interface as the output interface.

- b. (Optional) Allow configuration of filters on the destination interface for the named port-mirroring instance:

```
[edit forwarding-options port-mirroring family family output]
user@host# set no-filter-check
```

7. (Optional) Specify that any packets selected for mirroring are to be mirrored only once to any mirroring destination:

```
[edit forwarding-options port-mirroring family family output]
user@host# up 2
[edit forwarding-options port-mirroring]
user@host# set mirror-once
```



TIP: Enable the mirror-once option when an MX Series router is configured to perform Layer 2 port mirroring at both ingress and egress interfaces, which could result in sending duplicate packets to the same destination (which would complicate the analysis of the mirrored traffic).

8. Verify the minimum configuration of the global instance of Layer 2 port mirroring:

```
[edit forwarding-options ... ]
user@host# top
[edit]
user@host# show forwarding-options
```

```
forwarding-options {
  port-mirroring {
    input { # Global packet-selection properties.
      maximum-packet-length number; # Default is 0.
      rate number;
      run-length number;
    }
    family (ccc | vpls) { # Address- type 'bridge' displays as 'vpls'.
      output { # Global mirror destination properties.
        interface interface-name;
        no-filter-check; # Optional. Allow filters on interface.
      }
    }
    mirror-once; # Optional. Mirror destinations do not receive duplicate packets.
  }
}
```

**Related
Documentation**

- Layer 2 Port Mirroring Overview on page 85
- Layer 2 Port Mirroring Global Instance on page 87
- Examples: Layer 2 Port-Mirroring at Multiple Levels of the Chassis on page 127
- Example: Layer 2 Port Mirroring with Multiple Instances on page 129

Defining a Named Instance of Layer 2 Port Mirroring

On an MX Series router, you can define a set of Layer 2 port-mirroring properties that you can bind to a particular Packet Forwarding Engine (at the PIC level of the router chassis) or to group of Packet Forwarding Engines (at the DPC or FPC level of the route chassis).



NOTE: MX80 routers only support port mirroring at the global level (not for a particular instance).

To define a named instance of Layer 2 port mirroring on an MX Series router:

1. Enable configuration of a named instance of Layer 2 port mirroring :

```
[edit]
user@host# edit forwarding-options port-mirroring instance pm-instance-name
```

2. Enable configuration of the packet-sampling properties:

```
[edit forwarding-options port-mirroring instance pm-instance-name]
user@host# edit input
```

3. Specify packet-selection properties:

a. Specify the number of packets to select:

```
[edit forwarding-options port-mirroring instance pm-instance-name input]
user@host# set rate number
```

The valid range is 1 through 65535.

b. Specify the number of packets to mirror from each selection:

```
[edit forwarding-options port-mirroring instance pm-named-instance input]
user@host# set run-length number
```

The valid range is 0 through 20. The default value is 0.



NOTE: The `run-length` statement is not supported on MX80 routers.

c. Specify the length to which mirrored packets are to be truncated:

```
[edit forwarding-options port-mirroring instance pm-instance-name input]
user@host# set maximum-packet-length number
```

The valid range is 0 through 9216. The default value is 0, which means the mirrored packets are not truncated.



NOTE: The `maximum-packet-length` statement is not supported on MX80 routers.

4. Enable configuration of the mirror destination properties for Layer 2 packets that are part of bridging domain, Layer 2 switching cross-connects, or virtual private LAN service (VPLS):

a. Specify the Layer 2 address family type of traffic to be mirrored:

```
[edit forwarding-options port-mirroring instance pm-instance-name input]
user@host# up
[edit forwarding-options port-mirroring instance pm-instance-name]
user@host# edit family family
```

The value of the `family` option can be `bridge`, `ccc`, or `vpls`.



NOTE: Under the `[edit forwarding-options port-mirroring]` hierarchy level, the protocol family statement `family bridge` is an alias for `family vpls`. The command-line interface (CLI) displays Layer 2 port-mirroring configurations as `family vpls`, even for Layer 2 port-mirroring configured as `family bridge`. Use `family bridge` when the physical interface is configured with `encapsulation ethernet-bridge`.

b. Enable configuration of the mirror destination properties:

```
[edit forwarding-options port-mirroring instance pm-instance-name family family]
user@host# edit output
```

5. Specify mirror destination properties.

- a. Specify the physical interface on which to send the mirrored packets:

```
[edit forwarding-options port-mirroring instance pm-instance-name family family
output]
user@host# set interface interface-name
```

- b. (Optional) Allow configuration of filters on the destination interface for the global port-mirroring instance:

```
[edit forwarding-options port-mirroring instance pm-instance-name family family
output]
user@host# set no-filter-check
```



NOTE: You cannot configure port mirroring instances on MX80 routers. You can only configure port mirroring at the global level on MX80 routers.

6. (Optional) Specify that any packets selected for mirroring are to be mirrored only once to any mirroring destination:

```
[edit forwarding-options port-mirroring instance pm-instance-name family family
output]
user@host# up 3
[edit forwarding-options port-mirroring]
user@host# set mirror-once
```



TIP: Enable the global mirror-once option when an MX Series router is configured to perform Layer 2 port mirroring at both ingress and egress interfaces, which could result in sending duplicate packets to the same destination (which in turn would complicate the analysis of the mirrored traffic).

7. To configure a mirroring destination for a different packet family type, repeat steps 4 through 6.

8. Verify the minimum configuration of the named instances of Layer 2 port mirroring:

```
[edit forwarding-options ... ]
user@host# top
[edit]
user@host# show forwarding-options

forwarding-options {
  port-mirroring {
    ... optional-global-port-mirroring-configuration ...
    instance {
      pm-instance-name ( # A named instance of port mirroring
        input { # Packet-selection properties
```

```

        maximum-packet-length number; # Default is 0.
        rate number;
        run-length number;
    }
    family (ccc | vpls) { # Address- type 'bridge' displays as 'vpls'.
        output { # Mirror destination properties
            interface interface-name;
            no-filter-check; # Optional. Allow filters on interface.
        }
    }
}
}
}
mirror-once; # Optional. Mirror destinations do not receive duplicate packets.
}
}

```

Related Documentation

- Layer 2 Port Mirroring Overview on page 85
- Layer 2 Port Mirroring Named Instances on page 88
- Binding Layer 2 Port Mirroring to Ports Grouped at the FPC Level on page 108
- Binding Layer 2 Port Mirroring to Ports Grouped at the PIC Level on page 110
- Examples: Layer 2 Port-Mirroring at Multiple Levels of the Chassis on page 127
- Example: Layer 2 Port Mirroring with Multiple Instances on page 129

Displaying Information About DPCs or FPCs in an MX Series Router

To display information about the number and types of DPCs or FPCs in an MX Series router, the number of Packet Forwarding Engines on each, and the number and types of ports per Packet Forwarding Engine, you can use the following chassis operational mode commands:

- **show chassis hardware**
- **show chassis fabric fpcs**

For more information about chassis operational mode commands, see the [Junos OS System Basics and Services Command Reference](#).

Related Documentation

- Layer 2 Port Mirroring Overview on page 85
- Layer 2 Port Mirroring Properties on page 86
- Layer 2 Port Mirroring Named Instances on page 88

Binding Layer 2 Port Mirroring to Ports Grouped at the FPC Level

On an MX Series router, you can bind a named instance of Layer 2 port mirroring to a specific DPC or to a specific FPC in the router chassis. This is known as binding a named instance of Layer 2 port mirroring *at the FPC level* of the router chassis. The port mirroring properties specified in the named instance are applied to all physical ports associated with all Packet Forwarding Engines on the specified DPC or FPC.



NOTE: You can also bind a named instance of Layer 2 port mirroring to a specific Packet Forwarding Engine on a DPC or FPC in the router chassis.

For any packet-type family supported by Layer 2 port mirroring

- Port mirroring properties bound to a specific DPC or FPC override any port-mirroring properties configured at the global level.
- Port mirroring properties bound to a specific Packet Forwarding Engine override any port-mirroring properties configured at the DPC or FPC level.

You can apply up to two named instances of Layer 2 port mirroring to the same group of ports within the router chassis. By applying two different port-mirroring instances to the same DPC or FPC, you can bind two distinct Layer 2 port mirroring specifications to a single group of ports.

Before you begin, complete the following tasks:

- Define a named instance of Layer 2 port mirroring. See “Defining a Named Instance of Layer 2 Port Mirroring” on page 104.
- Display information about the number and types of DPCs or FPCs in the MX Series router, the number of Packet Forwarding Engines on each, and the number and types of ports per Packet Forwarding Engine. See “Displaying Information About DPCs or FPCs in an MX Series Router” on page 107.

To bind a named instance of Layer 2 port-mirroring to a DPC or FPC and its Packet Forwarding Engines:

1. Enable configuration of the router chassis properties:

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

2. Enable configuration of a DPC (and its corresponding Packet Forwarding Engines) or an FPC (and its installed PICs):

```
[edit chassis]
user@host# edit fpc slot-number
```

3. Bind a named instance of Layer 2 port mirroring (*pm-instance-name*) to the DPC or FPC:

```
[edit chassis fpc slot-number]
user@host# set port-mirror-instance pm-instance-name
```


4. (Optional) To bind a second named instance of Layer 2 port mirroring to the same DPC or FPC, repeat step 3 and specify a different named instance of Layer 2 port mirroring.
5. Verify the minimum configuration of the binding:

```
[edit chassis fpc slot-number port-mirror-instance pm-instance-name]
user@host# top
[edit]
user@host# show chassis

chassis {
  fpc slot-number { # Bind two port mirroring named instances at the FPC level.
    port-mirror-instance pm-instance-name-1;
    port-mirror-instance pm-instance-name-2;
  }
}
```

**Related
Documentation**

- Layer 2 Port Mirroring Overview on page 85
- Layer 2 Port Mirroring Named Instances on page 88
- Defining a Named Instance of Layer 2 Port Mirroring on page 104
- Binding Layer 2 Port Mirroring to Ports Grouped at the PIC Level on page 110
- Examples: Layer 2 Port-Mirroring at Multiple Levels of the Chassis on page 127
- Example: Layer 2 Port Mirroring with Multiple Instances on page 129

Binding Layer 2 Port Mirroring to Ports Grouped at the PIC Level

On an MX Series router, you can bind a named instance of Layer 2 port mirroring to the ports associated with a specific Packet Forwarding Engine (on a DPC) or to the ports associated with a specific PIC (installed in an FPC). This is known as binding a named instance of Layer 2 port mirroring *at the PIC level* of the router chassis. The port-mirroring properties specified in the named instance are applied to all physical ports associated with the specified Packet Forwarding Engine.



NOTE: You can also bind a named instance of Layer 2 port mirroring to a specific DPC or FPC in the router chassis.

For any packet-type family supported by Layer 2 port mirroring:

- Port mirroring properties bound to a specific Packet Forwarding Engine override any port-mirroring properties configured at the DPC or FPC level.
- Port mirroring properties bound to a specific DPC or FPC override any port-mirroring properties configured at the global level.

You can apply up to two named instances of Layer 2 port mirroring to the same group of ports within the router chassis. By applying two different port-mirroring instances to the same Packet Forwarding Engine or PIC, you can bind two distinct Layer 2 port mirroring specifications to a single group of ports.

For MX960 routers, there is a one-to-one mapping of Packet Forwarding Engines to Ethernet ports. Therefore, on MX960 routers only, you can bind a named instance of Layer 2 port mirroring to a *specific port* by binding the instance to the Packet Forwarding Engine associated with the port.

Before you begin, complete the following tasks:

- Define a named instance of Layer 2 port mirroring. See “Defining a Named Instance of Layer 2 Port Mirroring” on page 104.
- Display information about the number and types of DPCs in the MX Series router, the number of Packet Forwarding Engines on each DPC, and the number and types of ports per Packet Forwarding Engine. See “Displaying Information About DPCs or FPCs in an MX Series Router” on page 107.

To bind a named instance of Layer 2 port-mirroring to a Packet Forwarding Engine:

1. Enable configuration of the router chassis properties:

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

2. Enable configuration of a Packet Forwarding Engine or PIC:

```
[edit chassis]
user@host# edit fpc slot-number
user@host# edit pic slot-number
```

3. Bind a named instance of Layer 2 port mirroring (*pm-instance-name*) to the Packet Forwarding Engine or PIC:

```
[edit chassis fpc slot-number pic slot-number]
user@host# set port-mirror-instance pm-instance-name
```

4. (Optional) To bind a second named instance of Layer 2 port mirroring to the same Packet Forwarding Engine or PIC, repeat step 3 and specify a different named instance of Layer 2 port mirroring.
5. Verify the minimum configuration of the binding:

```
[edit forwarding-options ... ]
user@host# top
[edit]
user@host# show chassis
chassis {
  fpc slot-number {
    ... optional-binding-of-a-port-mirroring-instance-at-the-dpc-level ...
    pic slot-number { # Bind two port-mirroring named instances at the PIC level.
      port-mirror-instance pm-instance-name-1;
      port-mirror-instance pm-instance-name-2;
    }
  }
}
```

Related Documentation

- Layer 2 Port Mirroring Overview on page 85
- Layer 2 Port Mirroring Named Instances on page 88
- Defining a Named Instance of Layer 2 Port Mirroring on page 104
- Binding Layer 2 Port Mirroring to Ports Grouped at the FPC Level on page 108
- Examples: Layer 2 Port-Mirroring at Multiple Levels of the Chassis on page 127
- Example: Layer 2 Port Mirroring with Multiple Instances on page 129

Displaying Layer 2 Port-Mirroring Instance Settings and Status

To display the current state of port-mirroring instances, use the **show forwarding-options port-mirroring <terse | detail> <instance-name>** operational command.

For more information about displaying port mirroring instance settings and status, see the [Junos OS System Basics Configuration Guide](#).

Related Documentation

- Layer 2 Port Mirroring Overview on page 85
- Layer 2 Port Mirroring Global Instance on page 87
- Layer 2 Port Mirroring Named Instances on page 88
- Configuring the Global Instance of Layer 2 Port Mirroring on page 101
- Defining a Named Instance of Layer 2 Port Mirroring on page 104
- Disabling Layer 2 Port Mirroring Instances on page 112

- Examples: Layer 2 Port-Mirroring at Multiple Levels of the Chassis on page 127
- Example: Layer 2 Port Mirroring with Multiple Instances on page 129

Disabling Layer 2 Port Mirroring Instances

You can disable the global instance of Layer 2 port mirroring, a particular named instance, or all instances of port mirroring:

- To disable the global instance of Layer 2 port mirroring, include the **disable** statement at the **[edit forwarding-options port-mirroring]** hierarchy level:

```
[edit]
forwarding-options {
  port-mirroring {
    disable; Disables the global instance of Layer 2 port mirroring.
    ...global-instance-of-layer-2-port-mirroring-configuration...
  }
}
```

- To disable the definition of a particular named instance of Layer 2 port mirroring, include the **disable** statement at the **[edit forwarding-options port-mirroring instance instance-name]** hierarchy level:

```
[edit]
forwarding-options {
  port-mirroring {
    ...optional-configuration-of-the-global-instance-of-layer-2-port-mirroring...
    instance {
      port-mirroring-instance-name {
        disable; Disables this named instance of Layer 2 port mirroring.
        ...definition-of-a-named-instance-of-layer-2-port-mirroring...
      }
    }
  }
}
```

- To disable the global instance and all named instances of Layer 2 port mirroring, include the **disable-all-instances** statement at the **[edit forwarding-options port-mirroring]** hierarchy level:

```
[edit]
forwarding-options {
  port-mirroring {
    disable-all-instances; Disables all instances of Layer 2 port mirroring.
    ...optional-configuration-of-the-global-instance-of-layer-2-port-mirroring...
    instance {
      port-mirroring-instance-name {
        ...definition-of-a-named-instance-of-layer-2-port-mirroring...
      }
    }
  }
}
```

Related Documentation

- Layer 2 Port Mirroring Overview on page 85

- Layer 2 Port Mirroring Global Instance on page 87
- Layer 2 Port Mirroring Named Instances on page 88
- Displaying Layer 2 Port-Mirroring Instance Settings and Status on page 111

Configuring Layer 2 Port Mirroring for Logical Interfaces

The following topics describe the configuration of Layer 2 port mirroring for logical interfaces in MX Series routers:

- Defining a Layer 2 Port-Mirroring Firewall Filter on page 113
- Applying Layer 2 Port Mirroring to a Logical Interface on page 117
- Applying Layer 2 Port Mirroring to Traffic Forwarded or Flooded to a Bridge Domain on page 120
- Applying Layer 2 Port Mirroring to Traffic Forwarded or Flooded to a VPLS Routing Instance on page 122

Defining a Layer 2 Port-Mirroring Firewall Filter

For virtual private LAN service (VPLS) traffic (**family bridge** or **family vpls**) and for Layer 2 VPNs with **family ccc** on MX Series routers only, you can define a firewall filter that specifies Layer 2 port mirroring as the action to be performed if a packet matches the conditions configured in the firewall filter term.

You can use a Layer 2 port-mirroring firewall filter in the following ways:

- To mirror packets received or sent on a logical interface.
- To mirror packets forwarded or flooded to a bridge domain.
- To mirror packets forwarded or flooded to a VPLS routing instance.
- To mirror tunnel interface input packets only to multiple destinations.

For a summary of the three types of Layer 2 port-mirroring you can configure on an MX Series router, see “Application of Layer 2 Port Mirroring Types” on page 94.

For information about configuring firewall filters in general (including in a Layer 3 environment), see Stateless Firewall Filter Overview and How Standard Firewall Filters Evaluate Packets in the *Junos OS Firewall Filter and Policers Configuration Guide*.

To define a firewall filter with a Layer 2 port-mirroring action:

1. Enable configuration of firewall filters for Layer 2 packets that are part of a bridge domain, a Layer 2 switching cross-connect, or a virtual private LAN service (VPLS):

```
[edit]
user@host# edit firewall family family
```

The value of the **family** option can be **bridge**, **ccc**, or **vpls**.

2. Enable configuration of a firewall filter *pm-filter-name*:

```
[edit firewall family family]  
user@host# edit filter pm-filter-name
```

3. Enable configuration of a firewall filter term *pm-filter-term-name*:

```
[edit firewall family family filter pm-filter-name]  
user@host# edit term pm-filter-term-name
```

For more information about firewall filter terms in general (including in a Layer 3 environment), see Guidelines for Configuring Standard Firewall Filters in the [Junos OS Firewall Filter and Policer Configuration Guide](#).

4. (Optional) Specify the firewall filter match conditions based on the route source address *only if* you want to mirror a subset of the sampled packets.

For information about configuring firewall filter match conditions in general (including in a Layer 3 environment), see Firewall Filter Match Conditions Based on Numbers or Text Aliases, Firewall Filter Match Conditions Based on Bit-Field Values, Firewall Filter Match Conditions Based on Address Fields, and Firewall Filter Match Conditions Based on Address Classes, in the [Junos OS Firewall Filter and Policer Configuration Guide](#).

- For detailed information about Layer 2 bridging firewall filter match conditions (which are supported on MX Series routers only), see Standard Firewall Filter Match Conditions for Layer 2 Bridging Traffic.
- For detailed information about VPLS firewall filter match conditions, see Standard Firewall Filter Match Conditions for VPLS Traffic.
- For detailed information about Layer 2 circuit cross-connect (CCC) firewall filter match conditions, see Standard Firewall Filter Match Conditions for Layer 2 CCC Traffic.



NOTE: If you want all sampled packets to be considered to match (and be subjected to the actions specified in the **then** statement), then omit the **from** statement altogether.

5. Enable configuration of the **action** and **action-modifier** to apply to matching packets:

```
[edit firewall family family filter pm-filter-name term pm-filter-term-name]  
user@host# edit then
```

6. Specify the actions to be taken on matching packets:

```
[edit firewall family family filter pm-filter-name term pm-filter-term-name then]  
user@host# set action
```

The recommended value for the **action** is **accept**. If you do not specify an action, or if you omit the **then** statement entirely, all packets that match the conditions in the **from** statement are accepted.

7. Specify Layer 2 port mirroring or a next-hop group as the *action-modifier*:

- To reference the Layer 2 port mirroring properties currently in effect for the Packet Forwarding Engine or PIC associated with the underlying physical interface, use the **port-mirror** statement:

```
[edit firewall family family filter pm-filter-name term pm-filter-term-name then]
user@host# set port-mirror
```

- To reference the Layer 2 port mirroring properties configured in a specific named instance, use the **port-mirror-instance** *pm-instance-name* action modifier:

```
[edit firewall family family filter pm-filter-name term pm-filter-term-name then]
user@host# set port-mirror-instance pm-instance-name
```

If the underlying physical interface is not bound to a named instance of Layer 2 port mirroring but instead is implicitly bound to the global instance of Layer 2 port mirroring, then traffic at the logical interface is mirrored according to the properties specified in the named instance referenced by the **port-mirror-instance** action modifier.

- To reference a next-hop group that specifies the next-hop addresses (for sending additional copies of packets to an analyzer), use the **next-hop-group** *pm-next-hop-group-name* action modifier:

```
[edit firewall family family filter pm-filter-name term pm-filter-term-name then]
user@host# set next-hop-group pm-next-hop-group-name
```

For configuration information about next-hop groups, see “Defining a Next-Hop Group for Layer 2 Port Mirroring” on page 124. If you specify a next-hop group for Layer 2 port mirroring, the firewall filter term applies to the tunnel interface input only.

8. Verify the minimum configuration of the Layer 2 port-mirroring firewall filter:

```
[edit firewall ... ]
user@host# top
[edit]
user@host# show firewall
```

```
family (bridge | ccc | vpls) { # Type of packets to mirror
  filter pm-filter-name { # Firewall filter name
    term pm-filter-term-name {
      from { # Do not specify match conditions based on route source address
      }
      then {
        action; # Recommended action is 'accept'
        action-modifier; # Three options for Layer 2 port mirroring
      }
    }
  }
}
```

In the firewall filter term **then** statement, the *action-modifier* can be **port-mirror**, **port-mirror-instance** *pm-instance-name*, or **next-hop-group** *pm-next-hop-group-name*.

**Related
Documentation**

- [Layer 2 Port Mirroring Overview on page 85](#)
- [Layer 2 Port Mirroring Firewall Filters on page 90](#)
- [Layer 2 Port Mirroring to Multiple Destinations Using Next-Hop Groups on page 92](#)
- [Example: Layer 2 Port Mirroring at a Logical Interface on page 133](#)
- [Example: Layer 2 Port Mirroring for a Layer 2 VPN on page 135](#)
- [Example: Layer 2 Port Mirroring for a Layer 2 VPN with LAG Links on page 138](#)
- [Example: Layer 2 Port Mirroring to Multiple Destinations on page 140](#)

Applying Layer 2 Port Mirroring to a Logical Interface

You can apply a Layer 2 port-mirroring firewall filter to the input or to the output of a logical interface, including an aggregated Ethernet logical interface. Only packets of the address-type family specified by the filter action are mirrored.

Before you begin, complete the following task:

- Define a Layer 2 port-mirroring firewall filter to be applied to the input to a logical interface or output to a logical interface. For details, see “Defining a Layer 2 Port-Mirroring Firewall Filter” on page 113.



NOTE: This configuration task shows two Layer 2 port-mirroring firewall filters: one filter applied to the logical interface ingress traffic, and one filter applied to the logical interface egress traffic.

To apply a Layer 2 port-mirroring firewall filter to an input or output logical interface:

1. Configure the underlying physical interface for the logical interface.

- a. Enable configuration of the underlying physical interface:

```
[edit]
user@host# edit interfaces interface-name
```



NOTE: A port-mirroring firewall filter can also be applied to an aggregated-Ethernet logical interface.

- b. For Fast Ethernet and Gigabit Ethernet interfaces and aggregated Ethernet interfaces configured for VPLS, enable the reception and transmission of 802.1Q VLAN-tagged frames on the interface:

```
[edit interfaces interface-name]
user@host# set vlan-tagging
```

- c. For Ethernet interfaces that have IEEE 802.1Q VLAN tagging and bridging enabled and that must accept packets carrying TPID 0x8100 or a user-defined TPID, set the logical link-layer encapsulation type:

```
[edit interfaces interface-name]
user@host# set encapsulation extended-vlan-bridge
```

2. Configure the logical interface to which you want to apply a Layer 2 port-mirroring firewall filter.

- a. Specify the logical unit number:

```
[edit interfaces interface-name]
user@host# edit unit logical-unit-number
```

- b.

For a Fast Ethernet, Gigabit Ethernet, or Aggregated Ethernet interface, bind an 802.1Q VLAN tag ID to the logical interface:

```
[edit interfaces interface-name unit logical-unit-number]
user@host# set vlan-id number
```

3. Enable specification of an input or output filter to be applied to Layer 2 packets that are part of bridging domain, Layer 2 switching cross-connects, or virtual private LAN service (VPLS).

- If the filter is to be evaluated when packets are received on the interface:

```
[edit interfaces interface-name unit logical-unit-number]
user@host# set family family filter input pm-filter-name-a
```

- If the filter is to be evaluated when packets are sent on the interface:

```
[edit interfaces interface-name unit logical-unit-number]
user@host# set family family filter output pm-filter-name-b
```

The value of the *family* option can be **bridge**, **ccc**, or **vpls**.



NOTE: If port-mirroring firewall filters are applied at both the input and output of a logical interface, two copies of each packet are mirrored. To prevent the router from forwarding duplicate packets to the same destination, include the optional mirror-once statement at the [edit forwarding-options] hierarchy level.

4. Verify the minimum configuration for applying a named Layer 2 port mirroring firewall filter to a logical interface:

```
[edit interfaces interface-name unit logical-unit-number family family filter ... ]
user@host# top
[edit]
user@host# show interfaces
```

```
interfaces {
  interface-name {
    vlan-tagging;
    encapsulation extended-vlan-bridge;
    unit number { # Apply a filter to the input of this interface
      vlan-id number;
      family (bridge | ccc | vpls) {
        filter {
          input pm-filter-for-logical-interface-input;
        }
      }
    }
    unit number { # Apply a filter to the output of this interface
      vlan-id number;
      family (bridge | ccc | vpls) {
        filter {
          output pm-filter-for-logical-interface-output;
        }
      }
    }
  }
}
```

```
    }  
  }  
}
```

**Related
Documentation**

- Layer 2 Port Mirroring Overview on page 85
- Layer 2 Port Mirroring Firewall Filters on page 90
- Defining a Layer 2 Port-Mirroring Firewall Filter on page 113
- Applying Layer 2 Port Mirroring to Traffic Forwarded or Flooded to a Bridge Domain on page 120
- Applying Layer 2 Port Mirroring to Traffic Forwarded or Flooded to a VPLS Routing Instance on page 122
- Example: Layer 2 Port Mirroring at a Logical Interface on page 133
- Example: Layer 2 Port Mirroring for a Layer 2 VPN on page 135
- Example: Layer 2 Port Mirroring for a Layer 2 VPN with LAG Links on page 138

Applying Layer 2 Port Mirroring to Traffic Forwarded or Flooded to a Bridge Domain

You can apply a Layer 2 port-mirroring firewall filter to traffic being forwarded or flooded to a bridge domain. Only packets of the specified family type and forwarded or flooded to that bridge domain are mirrored.

Before you begin, complete the following task:

- Define a Layer 2 port-mirroring firewall filter to be applied to the traffic being forwarded to a bridge domain or flooded to a bridge domain. For details, see “Defining a Layer 2 Port-Mirroring Firewall Filter” on page 113.



NOTE: This configuration task shows two Layer 2 port-mirroring firewall filters: one filter applied to the bridge domain forwarding table ingress traffic, and one filter applied to the bridge domain flood table ingress traffic.

To apply a Layer 2 port-mirroring firewall filter to the forwarding table or flood table of a bridge domain:

1. Enable configuration of the bridge domain **bridge-domain-name** to which you want to apply a Layer 2 port-mirroring firewall filter for forwarded or flooded traffic:

- For a bridge domain:

```
[edit]
user@host# edit bridge-domains bridge-domain-name
```

- For a bridge domain under a routing instance:

```
[edit]
user@host# edit routing-instances routing-instance-name bridge-domains
bridge-domain-name
user@host# set instance-type vpls
```

For more detailed configuration information, see “Configuring a VPLS Routing Instance” on page 16.

2. Configure the bridge domain:

```
[edit]
user@host# set domain-type bridge
user@host# set interface interface-name
user@host# set routing-interface routing-interface-name
```

For more detailed configuration information, see “Configuring a Bridge Domain” on page 151 and “Configuring VLAN Identifiers for Bridge Domains and VPLS Routing Instances” on page 152.

3. Enable configuration of traffic forwarding on the bridge domain:

```
[edit ... bridge-domains bridge-domain-name]
user@host# edit forwarding-options
```

4. Apply a Layer 2 port-mirroring firewall filter to the bridge domain forwarding table or flood table.

- To mirror packets being forwarded to the bridge domain:

```
[edit ... bridge-domains bridge-domain-name forwarding-options]
user@host# set filter input pm-filter-for-bd-ingress-forwarded
```

- To mirror packets being flooded to the bridge domain:

```
[edit ... bridge-domains bridge-domain-name forwarding-options]
user@host# set flood input pm-filter-for-bd-ingress-flooded
```

5. Verify the minimum configuration for applying a Layer 2 port-mirroring firewall filter to the forwarding table or flood table of the bridge domain.

- a. Navigate to the hierarchy level at which the bridge domain is configured:

- **[edit]**
- **[edit routing-instances *routing-instance-name*]**

- b. Display the bridge domain configurations:

```
user@host# show bridge domains

bridge-domains {
  bridge-domain-name {
    instance-type vpls; # For a bridge domain under a routing instance.
    domain-type bridge;
    interface interface-name;
    forwarding-options {
      filter { # Mirror ingress forwarded traffic
        input pm-filter-for-bd-ingress-forwarded;
      }
      flood { # Mirror ingress flooded traffic
        input pm-filter-for-bd-ingress-flooded;
      }
    }
  }
}
```

Related Documentation

- Layer 2 Port Mirroring Overview on page 85
- Layer 2 Port Mirroring Firewall Filters on page 90
- Defining a Layer 2 Port-Mirroring Firewall Filter on page 113
- Applying Layer 2 Port Mirroring to a Logical Interface on page 117
- Applying Layer 2 Port Mirroring to Traffic Forwarded or Flooded to a VPLS Routing Instance on page 122
- Example: Layer 2 Port Mirroring at a Logical Interface on page 133
- Example: Layer 2 Port Mirroring for a Layer 2 VPN on page 135
- Example: Layer 2 Port Mirroring for a Layer 2 VPN with LAG Links on page 138

Applying Layer 2 Port Mirroring to Traffic Forwarded or Flooded to a VPLS Routing Instance

You can apply a Layer 2 port-mirroring firewall filter to traffic being forwarded or flooded to a VPLS routing instance. Only packets of the specified family type and forwarded or flooded to that VPLS routing instance are mirrored.

Before you begin, complete the following task:

- Define a Layer 2 port-mirroring firewall filter to be applied to the traffic being forwarded to a VPLS routing instance or flooded to a bridge domain. For details, see “Defining a Layer 2 Port-Mirroring Firewall Filter” on page 113.



NOTE: This configuration task shows two Layer 2 port-mirroring firewall filters: one filter applied to the VPLS routing instance forwarding table ingress traffic, and one filter applied to the VPLS routing instance flood table ingress traffic.

To apply a Layer 2 port-mirroring firewall filter to the forwarding table or flood table of a VPLS routing instance:

1. Enable configuration of the VPLS routing instance to which you want to apply a Layer 2 port-mirroring firewall filter for forwarded or flooded traffic:

```
[edit]
user@host# edit routing-instances routing-instance-name
user@host# set instance-type vpls
user@host# set interface interface-name
user@host# set route-distinguisher (as-number:number | ip-address:number)
user@host# set vrf-import [policy-names]
user@host# set vrf-export [policy-names]
user@host# edit protocols vpls
user@host@ ... vpls-configuration ...
```

For more detailed configuration information, see “Configuring a VPLS Routing Instance” on page 16.

2. Enable configuration of traffic forwarding on the VPLS routing instance:

```
[edit routing-instances routing-instance-name protocols vpls]
user@host# up 2
[edit routing-instances routing-instance-name]
user@host# edit forwarding-options
```

3. Apply a Layer 2 port-mirroring firewall filter to the VPLS routing instance forwarding table or flood table.

- To mirror packets being forwarded to the VPLS routing instance:

```
[edit routing-instances routing-instance-name forwarding-options]
user@host# set filter input pm-filter-for-vpls-ri-forwarded
```

- To mirror packets being flooded to the VPLS routing instance:

```
[edit routing-instances routing-instance-name forwarding-options]
```

```
user@host# set flood input pm-filter-for-vpls-ri-flooded
```

4. Verify the minimum configuration for applying a Layer 2 port-mirroring firewall filter to the forwarding table or flood table of the VPLS routing instance:

```
[edit routing-instances routing-instance-name forwarding-options]
user@host# top
[edit]
user@host# show routing-instances
```

```
routing-instances {
  routing-instance-name {
    instance-type vpls;
    interface interface-name;
    route-distinguisher (as-number:number | ip-address:number);
    vrf-import [policy-names];
    vrf-export [policy-names];
    protocols {
      vpls {
        ...vpls-configuration...
      }
    }
    forwarding-options {
      family vpls {
        filter { # Mirror ingress forwarded traffic
          input pm-filter-for-vpls-ri-forwarded;
        }
        flood { # Mirror ingress flooded traffic
          input pm-filter-for-vpls-ri-flooded;
        }
      }
    }
  }
}
```

Related Documentation

- Layer 2 Port Mirroring Overview on page 85
- Layer 2 Port Mirroring Firewall Filters on page 90
- Defining a Layer 2 Port-Mirroring Firewall Filter on page 113
- Applying Layer 2 Port Mirroring to a Logical Interface on page 117
- Applying Layer 2 Port Mirroring to Traffic Forwarded or Flooded to a Bridge Domain on page 120
- Example: Layer 2 Port Mirroring at a Logical Interface on page 133
- Example: Layer 2 Port Mirroring for a Layer 2 VPN on page 135
- Example: Layer 2 Port Mirroring for a Layer 2 VPN with LAG Links on page 138

Configuring Layer 2 Port Mirroring to Multiple Destinations

The following topics describe the configuration of Layer 2 port mirroring to multiple destinations:

- Defining a Next-Hop Group for Layer 2 Port Mirroring on page 124
- Displaying Next-Hop Group Settings and Status on page 125

Defining a Next-Hop Group for Layer 2 Port Mirroring

On MX Series routers, you can mirror tunnel interface input traffic to multiple destinations. To this form of *multipacket port mirroring*, you specify two or more additional destinations in a next-hop group, define a firewall filter that references the next-hop group as the filter action, and then apply the filter to a logical tunnel interface (**lt-**) or virtual tunnel interface (**vt-**) on the MX Series router.



NOTE: This topic describes how to define a next-hop group for Layer 2 port mirroring to multiple destinations. For detailed information about defining a firewall filter for Layer 2 port mirroring to multiple destinations, see “Defining a Layer 2 Port-Mirroring Firewall Filter” on page 113.

To define a next-hop group for a Layer 2 port-mirroring firewall filter action:

1. Enable configuration of Layer 2 forwarding options.

- To enable Layer 2 forwarding options at the top level:

```
[edit]
user@host edit forwarding-options port-mirroring family (ccc | vpls) output
```

- To enable Layer 2 forwarding options for a routing instance:

```
[edit]
user@host edit forwarding-options port-mirroring instance instance-name
family (ccc | vpls) output
```

2. Enable configuration of a next-hop-group for Layer 2 port mirroring:

```
[edit forwarding-options port-mirroring ... family (ccc | vpls) output]
user@host# edit next-hop-group pm-next-hop-group-name
```

3. Specify the type of addresses to be used in the next-hop group configuration. By default, the next-hop group is specified using Layer 3 addresses (**group-type inet**). To specify the next-hop group using Layer 2 addresses instead, you must include the **group-type layer-2** statement:

```
[edit forwarding-options port-mirroring ... family (ccc | vpls) output next-hop-group
pm-next-hop-group-name]
user@host# set group-type layer-2
```


- Specify the logical interfaces of the next-hop router:

```
[edit forwarding-options port-mirroring ... family (ccc | vpls) output next-hop-group
  pm-next-hop-group-name]
user@host# set interface logical-interface-name-1
user@host# set interface logical-interface-name-2
```

The MX Series router supports up to 30 next-hop groups. Each next-hop group supports up to 16 next-hop addresses. Each next-hop group must specify at least two addresses.

- Verify the configuration of the next-hop group:

```
[edit forwarding-options port-mirroring ... family (ccc | vpls) output next-hop-group
  pm-next-hop-group-name]
user@host# top
[edit]
user@host# show forwarding-options

...
next-hop-group pm-next-hop-group-name { # Next-hop group on a bridge domain.
  group-type layer-2;
  interface logical-interface-name-1;
  interface logical-interface-name-2;
}
...
```

Related Documentation

- Layer 2 Port Mirroring Overview on page 85
- Layer 2 Port Mirroring to Multiple Destinations Using Next-Hop Groups on page 92
- Defining a Layer 2 Port-Mirroring Firewall Filter on page 113
- Displaying Next-Hop Group Settings and Status on page 125
- Example: Layer 2 Port Mirroring to Multiple Destinations on page 140

Displaying Next-Hop Group Settings and Status

To display the current state of next-hop groups, use the **show forwarding-options next-hop-group** <terse | brief | detail> <group-name> operational command.

For more information, see the [Junos OS System Basics and Services Command Reference](#).

Related Documentation

- Layer 2 Port Mirroring Overview on page 85
- Layer 2 Port Mirroring to Multiple Destinations Using Next-Hop Groups on page 92
- Defining a Layer 2 Port-Mirroring Firewall Filter on page 113
- Defining a Next-Hop Group for Layer 2 Port Mirroring on page 124
- Example: Layer 2 Port Mirroring to Multiple Destinations on page 140

CHAPTER 10

Examples of Layer 2 Port Mirroring

- Examples: Layer 2 Port-Mirroring at Multiple Levels of the Chassis on page 127
- Example: Layer 2 Port Mirroring with Multiple Instances on page 129
- Example: Layer 2 Port Mirroring at a Logical Interface on page 133
- Example: Layer 2 Port Mirroring for a Layer 2 VPN on page 135
- Example: Layer 2 Port Mirroring for a Layer 2 VPN with LAG Links on page 138
- Example: Layer 2 Port Mirroring to Multiple Destinations on page 140

Examples: Layer 2 Port-Mirroring at Multiple Levels of the Chassis

On an MX Series router, you can apply named instances of Layer 2 port mirroring at the FPC or DPC level of the chassis or at the PIC level of the chassis. However, you can configure (and implicitly apply) only one global instance of Layer 2 port mirroring to the entire chassis.

- Layer 2 Port Mirroring at the FPC Level on page 127
- Layer 2 Port Mirroring at the PIC Level on page 128
- Layer 2 Port Mirroring at the FPC and PIC Levels on page 128

Layer 2 Port Mirroring at the FPC Level

In this example configuration of an MX Series router chassis, a named instance of Layer 2 port mirroring (**pm1**) is bound to physical ports grouped at the FPC level:

```
[edit]
chassis {
  fpc 2 {
    port-mirror-instance pm1;
  }
}
```

This is not a complete configuration. The physical interfaces associated with the FPC or DPC in slot 2 must be configured at the **[edit interfaces]** hierarchy level. The Layer 2 port mirroring named instance **pm1** must be configured at the **[edit forwarding-options port-mirroring instance]** hierarchy level.

Layer 2 Port Mirroring at the PIC Level

In this example configuration of an MX Series router chassis, a named instance of Layer 2 port mirroring (**pm2**) is bound to the physical ports grouped at the PIC level:

```
[edit]
chassis {
  fpc 2 {
    pic 0 {
      port-mirror-instance pm2;
    }
  }
}
```

This is not a complete configuration. The physical interfaces associated with the FPC or DPC in slot 2 must be configured at the **[edit interfaces]** hierarchy level. The Layer 2 port mirroring named instance **pm2** must be configured at the **[edit forwarding-options port-mirroring instance]** hierarchy level.

Layer 2 Port Mirroring at the FPC and PIC Levels

In this example configuration of an MX Series router chassis, one named instance of Layer 2 port mirroring (**pm1**) is applied at the FPC level of the router chassis. A second named instance (**pm2**) is applied at the PIC level:

```
[edit]
chassis {
  fpc 2 {
    port-mirror-instance pm1;
    pic 0 {
      port-mirror-instance pm2;
    }
  }
}
```

This is not a complete configuration. Physical interfaces associated with the FPC or DPC in slot 2, including physical interfaces associated with **pic 0**, must be configured at the **[edit interfaces]** hierarchy level. The Layer 2 port mirroring named instances **pm1** and **pm2** must be configured at the **[edit forwarding-options port-mirroring instance]** hierarchy level.

Related Documentation

- Layer 2 Port Mirroring Overview on page 85
- Layer 2 Port Mirroring Global Instance on page 87
- Layer 2 Port Mirroring Named Instances on page 88
- Configuring the Global Instance of Layer 2 Port Mirroring on page 101
- Defining a Named Instance of Layer 2 Port Mirroring on page 104

Example: Layer 2 Port Mirroring with Multiple Instances

Because you can configure more than one port-mirroring instance, care is required when specifying which instance is meant. This topic contains the following information:

- Example: Configuring Multiple Instances of Layer 2 Port Mirroring on page 129
- Explicit Reference of a Port Mirroring Instance on page 131
- Implicit Reference of Port Mirroring on the Underlying Physical Interface on page 132

Example: Configuring Multiple Instances of Layer 2 Port Mirroring

This configuration example illustrates the configuration of Layer 2 port mirroring at the physical interfaces associated with FPC 2, PIC 0 and at two logical interfaces on one of those ports.

At the physical interface levels of the router chassis, two named instances of port mirroring are configured and then bound to the group of physical ports associated with FPC 2, PIC 0.

At two of the logical interfaces on physical interface **ge-2/0/1**, two Layer 2 port-mirroring firewall filters are applied to the input traffic. One filter *explicitly* references the port mirroring properties specified in one of the named instances of port mirroring. The other filter *implicitly* references the port mirroring properties in effect on the underlying physical interface **ge-2/0/1**.

The resulting configuration is an example of the relationships that can exist between multiple instances of Layer 2 port mirroring applied to an MX Series router.

1. Configure two named instances of Layer 2 port mirroring (**pm_instance_1** and **pm_instance_2**), and include mirror destination properties for bridge domain traffic (**family bridge**):

```
[edit]
forwarding-options {
  port-mirroring {
    instance {
      pm_instance_1 {
        input {
          ... packet-selection-properties-configuration ...
        }
        family bridge {
          ... mirror-destination-properties-configuration ...
        }
      }
      pm_instance_2 {
        input {
          ... packet-selection-properties-configuration ...
        }
        family bridge {
          ... mirror-destination-properties-configuration ...
        }
      }
    }
  }
}
```

```
}
}
```



NOTE: In this example, no global port-mirroring properties are configured on the router.

2. Apply the Layer 2 port mirroring instances to the same group of ports in the router chassis. In this example, the named instances of Layer 2 port mirroring are applied to the same group of physical interfaces specified at the PIC level of the chassis:

```
[edit]
chassis {
  fpc 2 {
    pic 0 {
      port-mirror-instance pm_instance_1;
      port-mirror-instance pm_instance_2;
    }
  }
}
```

Note that, in this example, two named instances of Layer 2 port mirroring are bound to the PIC level of the chassis at the same group of ports.

3. Configure two Layer 2 port-mirroring firewall filters, both for bridge-domain traffic and with one of the filters explicitly referencing one of the named instances of Layer 2 port mirroring:
 - Configure the filter **pm_filter_1** to use the Layer 2 port-mirroring properties configured in the named port-mirroring instance **pm_instance_1**. To refer to the Layer 2 port mirroring properties configured in a particular named instance of port mirroring, use the **port-mirror-instance *port-mirroring-instance-name*** statement.
 - Configure the filter **pm_filter_2** to use the Layer 2 port mirroring properties in effect on the underlying physical interface of the logical interface to which the filter is applied. To refer to the Layer 2 port mirroring properties in effect on the underlying physical interface, use the **port-mirror** statement. If two instances of port mirroring are bound to that port, then the firewall filter uses the first instance bound within the **[edit chassis fpc slot-number]** or **[edit chassis fpc slot-number pic slot-number]** hierarchy level.

```
[edit]
firewall {
  family bridge {
    filter pm_filter_1 {
      term pm {
        then port-mirror-instance pm_instance_1;
      }
    }
    filter pm_filter_2 {
      term pm {
        then port-mirror;
      }
    }
  }
}
```



NOTE: Because the `port-mirror` filter action modifier relies on the port-mirroring properties defined at the `[edit forwarding-options port-mirroring]` hierarchy level, the `port-mirror` filter action is not supported for logical systems.

4. Apply the two Layer 2 port-mirroring firewall filters to logical interfaces on interface `ge-2/0/1`:

```
[edit]
interfaces {
  ge-2/0/1 {
    flexible-vlan-tagging;
    encapsulation ethernet-bridge;
    unit 0 {
      vlan-id 201;
      family bridge {
        filter { # Explicitly references a named instance of port mirroring.
          input pm_filter_1;
        }
      }
    }
    unit 1 {
      vlan-id 202;
      family bridge {
        filter { # Implicitly references the underlying port mirroring.
          input pm_filter_2;
        }
      }
    }
  }
}
```

Explicit Reference of a Port Mirroring Instance

On logical interface `ge-2/0/1.0`, the `port-mirror-instance` statement explicitly references the Layer 2 port mirroring properties in the named instance `pm_instance_1`. In this example, the port mirroring properties specified in `pm_instance_1` remain in effect at logical interface `ge-2/0/1.0` under the following conditions:

- The firewall filter `pm_filter_1` remains configured (as shown in step 3).
- The named instance `pm_instance_1` remains configured (as shown in step 1).

Even if the named instance `pm_instance_1` is no longer configured or no longer bound to the router chassis at FPC 2, PIC 0, the port mirroring properties specified in `pm_instance_1` remain in effect at logical interface `fe-2/0/1.0` through firewall filter `pm_filter_1`.

Implicit Reference of Port Mirroring on the Underlying Physical Interface

On logical interface **ge-2/0/1.1**, the **port-mirror** statement implicitly references the Layer 2 port mirroring properties in effect at the underlying physical interface **ge-2/0/1**. In this example, the port mirroring properties specified in **pm_instance_2** remain in effect at the ports associated with FPC 2, PIC 0 under the following conditions:

- The firewall filter **pm_filter_2** remains configured (as shown in step 3).
- The named instance **pm_instance_2** remains configured (as shown in step 1).
- The named instance **pm_instance_2** remains bound to the router chassis at FPC 2, PIC 0 (as shown in step 2).

If you disable the named instance **pm_instance_2** or delete its binding to the physical ports associated with FPC 2, PIC 0, then—if global Layer 2 port mirroring properties had been configured—the global port mirroring properties would be used at logical interface **ge-2/0/1.1** through firewall filter **pm_filter_2**.



NOTE: There is a limitation to a Layer 2 port mirroring firewall filter in which you implicitly reference Layer 2 port mirroring properties by including the **port-mirror** statement. If multiple named instances of Layer 2 port mirroring are bound to the underlying physical interface, then only the first binding in the stanza (or the only binding) is used at the logical interface. This is done mainly for backward compatibility.

In the example above, filter **pmff_bd_filter_2** uses the **port-mirror** statement, and so the filter action uses the mirroring properties of the first port mirroring instance applied to the router chassis at the **[edit chassis fpc 2 pic 0]** hierarchy level, which is the instance **pm_instance_1**.

Related Documentation

- Layer 2 Port Mirroring Overview on page 85
- Layer 2 Port Mirroring Named Instances on page 88
- Layer 2 Port Mirroring Firewall Filters on page 90
- Defining a Named Instance of Layer 2 Port Mirroring on page 104
- Defining a Layer 2 Port-Mirroring Firewall Filter on page 113

Example: Layer 2 Port Mirroring at a Logical Interface

The following steps describe an example in which the global port-mirroring instance and a port-mirroring firewall filter are used to configure Layer 2 port mirroring for the input to a logical interface.

1. Configure the bridge domain **example-bd-with-analyzer**, which contains the external packet analyzer, and the bridge domain **example-bd-with-traffic**, which contains the source and destination of the Layer 2 traffic being mirrored:

```
[edit]
bridge-domains {
  example-bd-with-analyzer { # Contains an external traffic analyzer
    vlan-id 1000;
    interface ge-2/0/0.0; # External analyzer
  }
  example-bd-with-traffic { # Contains traffic input and output interfaces
    vlan-id 1000;
    interface ge-2/0/6.0; # Traffic input port
    interface ge-3/0/1.2; # Traffic output port
  }
}
```

Assume that logical interface **ge-2/0/0.0** is associated with an external traffic analyzer that is to receive port-mirrored packets. Assume that logical interfaces **ge-2/0/6.0** and **ge-3/0/1.2** will be traffic input and output ports, respectively.

2. Configure Layer 2 port-mirroring for the global instance, with the port-mirroring destination being the bridge domain interface associated with the external analyzer (logical interface **ge-2/0/0.0** on bridge domain **example-bd-with-analyzer**). Be sure to enable the option that allows filters to be applied to this port-mirroring destination:

```
[edit]
forwarding-options {
  port-mirroring {
    input {
      rate 10;
      run-length 5;
    }
    family bridge {
      output {
        interface ge-2/0/0.0; # Mirror packets to the external analyzer
        no-filter-check; # Allow filters on the mirror destination interface
      }
    }
  }
}
```

The **input** statement at the **[edit forwarding-options port-mirroring]** hierarchy level specifies that sampling begins every tenth packet and that each of the first five packets selected are to be mirrored.

The **output** statement at the **[edit forwarding-options port-mirroring family bridge]** hierarchy level specifies the output mirror interface for Layer 2 packets in a bridging environment:

- Logical interface **ge-2/0/0.0**, which is associated with the external packet analyzer, is configured as the port-mirroring destination.
- The optional **no-filter-check** statement allows filters to be configured on this destination interface.

3. Configure the Layer 2 port-mirroring firewall filter **example-bridge-pm-filter**:

```
[edit]
firewall {
  family bridge {
    filter example-bridge-pm-filter {
      term example-filter-terms {
        then {
          accept;
          port-mirror;
        }
      }
    }
  }
}
```

When this firewall filter is applied to the input or output of a logical interface for traffic in a bridging environment, Layer 2 port mirroring is performed according to the input packet-sampling properties and mirror destination properties configured for the Layer 2 port mirroring global instance. Because this firewall filter is configured with the single, default filter action **accept**, all packets selected by the **input** properties (**rate = 10** and **run-length = 5**) match this filter.

4. Configure the logical interfaces:

```
[edit]
interfaces {
  ge-2/0/0 { # Define the interface to the external analyzer
    encapsulation ethernet-bridge;
    unit 0 {
      family bridge;
    }
  }
  ge-2/0/6 { # Define the traffic input port
    flexible-vlan-tagging;
    encapsulation extended-vlan-bridge;
    unit 0 {
      vlan-id 100;
      family bridge {
        filter {
          input example-bridge-pm-filter; # Apply the port-mirroring firewall filter
        }
      }
    }
  }
  ge-3/0/1 { # Define the traffic output port
```

```

flexible-vlan-tagging;
encapsulation extended-vlan-bridge;
unit 2 {
    vlan-tags outer 10 inner 20;
    family bridge;
}
}
}

```

Packets received at logical interface **ge-2/0/6.0** on bridge domain **example-bd-with-traffic** are evaluated by the port-mirroring firewall filter **example-bridge-pm-filter**. The firewall filter acts on the input traffic according to the filter actions configured in the firewall filter itself plus the input packet-sampling properties and mirror destination properties configured in the global port-mirroring instance:

- All packets received at **ge-2/0/6.0** are forwarded to their (assumed) normal destination at logical interface **ge-3/0/1.2**.
- For every ten input packets, copies of the first five packets in that selection are forwarded to the external analyzer at logical interface **ge-0/0/0.0** in the other bridge domain, **example-bd-with-analyzer**.

If you configure the port-mirroring firewall filter **example-bridge-pm-filter** to take the **discard** action instead of the **accept** action, all original packets are discarded while copies of the packets selected using the global port-mirroring **input** properties are sent to the external analyzer.

Related Documentation

- Layer 2 Port Mirroring Overview on page 85
- Layer 2 Port Mirroring Firewall Filters on page 90
- Defining a Layer 2 Port-Mirroring Firewall Filter on page 113

Example: Layer 2 Port Mirroring for a Layer 2 VPN

The following example is not a complete configuration, but shows all the steps needed to configure port mirroring on an L2VPN using **family ccc**.

1. Configure the bridge domain **port-mirror-bd**, which contains the external packet analyzer:

```

[edit]
bridge-domains {
    port-mirror-bd { # Contains an external traffic analyzer
        interface ge-2/2/9.0; # External analyzer
    }
}

```

2. Configure the Layer 2 VPN CCC to connect logical interface **ge-2/0/1.0** and logical interface **ge-2/0/1.1**:

```

[edit]
protocols {

```

```

mpls {
  interface all;
}
connections {
  interface-switch if_switch {
    interface ge-2/0/1.0;
    interface ge-2/0/1.1;
  }
}
}

```

3. Configure Layer 2 port mirroring for the global instance, with the port-mirroring destination being the bridge domain interface associated with the external analyzer (logical interface **ge-2/2/9.0** on bridge domain **example-bd-with-analyzer**):

```

[edit]
forwarding-options {
  port-mirroring {
    input {
      rate 1;
      maximum-packet-length 200;
    }
    family ccc {
      output {
        interface ge-2/2/9.0; # Mirror packets to the external analyzer
      }
    }
    instance {
      inst1 {
        input {
          rate 1;
          maximum-packet-length 300;
        }
        family ccc {
          output {
            interface ge-2/2/9.0;
          }
        }
      }
    }
  }
}
}

```

4. Define the Layer 2 port-mirroring firewall filter **pm_filter_ccc** for **family ccc**:

```

[edit]
firewall {
  family ccc {
    filter pm_filter_ccc {
      term pm {
        then port-mirror;
      }
    }
  }
}
}

```

5. Apply the port mirror instance to the chassis:

```
[edit]
chassis {
  fpc 2 {
    port-mirror-instance inst1;
  }
}
```

6. Configure interface **ge-2/2/9** for the VLANs, and configure interface **ge-2/0/1** for port mirroring with the **pm_filter_ccc** firewall filter:

```
[edit]
interfaces {
  ge-2/2/9 {
    encapsulation ethernet-bridge;
    unit 0 {
      family bridge;
    }
  }
  ge-2/0/1 {
    vlan-tagging;
    encapsulation extended-vlan-ccc;
    unit 0 {
      vlan-id 10;
      family ccc {
        filter {
          input pm_filter_ccc;
        }
      }
    }
    unit 1 {
      vlan-id 20;
      family ccc {
        filter {
          output pm_filter_ccc;
        }
      }
    }
  }
}
```

Related Documentation

- Layer 2 Port Mirroring Overview on page 85
- Layer 2 Port Mirroring Firewall Filters on page 90
- Defining a Layer 2 Port-Mirroring Firewall Filter on page 113

Example: Layer 2 Port Mirroring for a Layer 2 VPN with LAG Links

The following example is not a complete configuration, but shows all the steps needed to configure port mirroring on an L2VPN using **family ccc** and aggregated Ethernet links.

1. Configure the bridge domain **port_mirror_bd**, which contains the external packet analyzer:

```
[edit]
bridge-domains {
  port_mirror_bd { # Contains an external traffic analyzer
    interface ge-2/2/8.0; # External analyzer
  }
}
```

2. Configure the Layer 2 VPN CCC to connect interface **ae0.0** and interface **ae0.1**:

```
[edit]
protocols {
  mpls {
    interface all;
  }
}
connections {
  interface-switch if_switch {
    interface ae0.0;
    interface ae0.1;
  }
}
```

3. Configure Layer 2 port mirroring for the global instance, with the port-mirroring destination being the bridge domain interface associated with the external analyzer (logical interface **ge-2/2/9.0** on bridge domain **example_bd_with_analyzer**):

```
[edit]
forwarding-options {
  port-mirroring {
    input {
      rate 1;
      maximum-packet-length 200;
    }
    family ccc {
      output {
        interface ge-2/2/8.0; # Mirror packets to the external analyzer
      }
    }
    instance {
      pm_instance_1 {
        input {
          rate 1;
          maximum-packet-length 300;
        }
        family ccc {
          output {
            interface ge-2/2/8.0;
          }
        }
      }
    }
  }
}
```

```

    {
  }
}
}
}

```

4. Configure the firewall filter **pm_ccc** for **family ccc**:

```

[edit]
firewall {
  family ccc {
    filter pm_ccc {
      term pm {
        then port-mirror;
      }
    }
  }
}

```

5. Apply the aggregated Ethernet interfaces and port mirror instance to the chassis:

```

[edit]
chassis {
  aggregated-devices {
    ethernet {
      device-count 10;
    }
  }
  fpc 2 {
    port-mirror-instance pm_instance_1;
  }
}

```

6. Configure interfaces **ae0** and **ge-2/0/2** (for aggregated Ethernet) and **ge-2/2/8** (for port mirroring) with the **pm_ccc** filter:

```

[edit]
interfaces {
  ae0 {
    vlan-tagging;
    encapsulation extended-vlan-ccc;
    unit 0 {
      vlan-id 10;
      family ccc {
        filter {
          input pm_ccc;
        }
      }
    }
    unit 1 {
      vlan-id 20;
      family ccc {
        filter {
          output pm_ccc;
        }
      }
    }
  }
}

```

```
}
ge-2/0/2 {
  gigeather-options {
    802.3ad ae0;
  }
}
ge-2/2/8 {
  encapsulation ethernet-bridge;
  unit 0 {
    family bridge;
  }
}
```

**Related
Documentation**

- Layer 2 Port Mirroring Overview on page 85
- Layer 2 Port Mirroring Firewall Filters on page 90
- Defining a Layer 2 Port-Mirroring Firewall Filter on page 113

Example: Layer 2 Port Mirroring to Multiple Destinations

On MX Series routers, you can mirror traffic to multiple destinations by configuring next-hop groups in Layer 2 port-mirroring firewall filters applied to tunnel interfaces.

1. Configure the chassis to support tunnel services at PIC 0 on FPC 2. This configuration includes two logical tunnel interfaces on FPC 2, PIC 0, port 10.

```
[edit]
chassis {
  fpc 2 {
    pic 0 {
      tunnel-services {
        bandwidth 1g;
      }
    }
  }
}
```

2. Configure the physical and logical interfaces for three bridge domains and one Layer 2 VPN CCC:

- Bridge domain **bd** will span logical interfaces **ge-2/0/1.0** and **ge-2/0/1.1**.
- Bridge domain **bd_next_hop_group** will span logical interfaces **ge-2/2/9.0** and **ge-2/0/2.0**.
- Bridge domain **bd_port_mirror** will use the logical tunnel interface **lt-2/0/10.2**.
- Layer 2 VPN CCC **if_switch** will connect logical interfaces **ge-2/0/1.2** and **lt-2/0/10.1**.

```
[edit]
interfaces {
  ge-2/0/1 {
    flexible-vlan-tagging;
    encapsulation flexible-ethernet-services;
```



```

    unit 0 { # An interface on bridge domain 'bd'.
        encapsulation vlan-bridge;
        vlan-id 200;
        family bridge {
            filter {
                input pm_bridge;
            }
        }
    }
    unit 1 { # An interface on bridge domain 'bd'.
        encapsulation vlan-bridge;
        vlan-id 201;
        family bridge {
            filter {
                input pm_bridge;
            }
        }
    }
    unit 2 {
        encapsulation vlan-ccc;
        vlan-id 1000;
    }
}
ge-2/0/2 { # For 'bd_next_hop_group'
    encapsulation ethernet-bridge;
    unit 0 {
        family bridge;
    }
}
lt-2/0/10 {
    unit 1 {
        encapsulation ethernet-ccc;
        peer-unit 2;
    }
    unit 2 {
        encapsulation ethernet-bridge;
        peer-unit 1;
        family bridge {
            filter {
                output redirect_to_nhg;
            }
        }
    }
}
ge-2/2/9 {
    encapsulation ethernet-bridge;
    unit 0 { # For 'bd_next_hop_group'
        family bridge;
    }
}
}

```

3. Configure the three bridge domains and the Layer 2 VPN switching CCC:

- Bridge domain **bd** spans logical interfaces **ge-2/0/1.0** and **ge-2/0/1.1**.
- Bridge domain **bd_next_hop_group** spans logical interfaces **ge-2/2/9.0** and **ge-2/0/2.0**.
- Bridge domain **bd_port_mirror** uses the logical tunnel interface **lt-2/0/10.2**.
- Layer 2 VPN CCC **if_switch** connects interfaces **ge-2/0/1.2** and **lt-2/0/10.1**.

```
[edit]
bridge-domains {
  bd {
    interface ge-2/0/1.0;
    interface ge-2/0/1.1;
  }
  bd_next_hop_group {
    interface ge-2/2/9.0;
    interface ge-2/0/2.0;
  }
  bd_port_mirror {
    interface lt-2/0/10.2;
  }
}
protocols {
  mpls {
    interface all;
  }
  connections {
    interface-switch if_switch {
      interface ge-2/0/1.2;
      interface lt-2/0/10.1;
    }
  }
}
```

For detailed information about configuring the CCC connection for Layer 2 switching cross-connects, see the [Junos OS MPLS Applications Configuration Guide](#).

4. Configure forwarding options:

- Configure global port mirroring properties to mirror **family vpls** traffic to an interface on the bridge domain **bd_port_mirror**.
- Configure the next-hop group **nhg_mirror_to_bd** to forward Layer 2 traffic to the bridge domain **bd_next_hop_group**.

Both of these forwarding options will be referenced by the port-mirroring firewall filter:

```
[edit]
forwarding-options {
  port-mirroring { # Global port mirroring properties.
    input {
      rate 1;
    }
    family vpls {
      output {
```

```

        interface lt-2/0/10.2; # Interface on 'bd_port_mirror' bridge domain.
        no-filter-check;
    }
}
next-hop-group nhg_mirror_to_bd { # Configure a next-hop group.
    group-type layer-2; # Specify 'layer-2' for Layer 2; default 'inet' is for Layer 3.
    interface ge-2/0/2.0; # Interface on 'bd_next_hop_group' bridge domain.
    interface ge-2/2/9.0; # Interface on 'bd_next_hop_group' bridge domain.
}
}

```

5. Configure two Layer 2 port-mirroring firewall filters for **family bridge** traffic:

- **filter_pm_bridge**—Sends all **family bridge** traffic to the global port mirroring destination.
- **filter_redirect_to_nhg**—Sends all **family bridge** traffic to the final next-hop group **nhg_mirror_to_bd**.

Layer 2 port-mirroring firewall filters for **family bridge** traffic applies to traffic on a physical interface configured with encapsulation **ethernet-bridge**.

```

[edit]
firewall {
    family bridge {
        filter filter_pm_bridge {
            term term_port_mirror {
                then port-mirror;
            }
        }
        filter filter_redirect_to_nhg {
            term term_nhg {
                then next-hop-group nhg_mirror_to_bd;
            }
        }
    }
}

```

Related Documentation

- Layer 2 Port Mirroring Overview on page 85
- Layer 2 Port Mirroring to Multiple Destinations Using Next-Hop Groups on page 92
- Defining a Layer 2 Port-Mirroring Firewall Filter on page 113
- Defining a Next-Hop Group for Layer 2 Port Mirroring on page 124
- Displaying Next-Hop Group Settings and Status on page 125

PART 4

Layer 2 Bridging

- [Layer 2 Bridging Overview on page 147](#)
- [Configuring Layer 2 Bridging on page 151](#)
- [Summary of Layer 2 Bridging Configuration Statements on page 175](#)

Layer 2 Bridging Overview

- Layer 2 Bridge Domains Overview on page 147
- Layer 2 Virtual Switches Overview on page 148
- Layer 2 Learning and Forwarding for Bridge Domains Overview on page 149
- Layer 2 Learning and Forwarding for Bridge Domains Functioning as Switches with Layer 2 Trunk Ports on page 149
- Guidelines for Configuring VLAN Identifiers for Bridge Domains and VPLS Routing Instances on page 150

Layer 2 Bridge Domains Overview

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

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

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

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

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

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

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

**Related
Documentation**

- Layer 2 Virtual Switches Overview on page 148
- Layer 2 Learning and Forwarding for Bridge Domains Overview on page 149
- Layer 2 Learning and Forwarding for Bridge Domains Functioning as Switches with Layer 2 Trunk Ports on page 149
- Configuring a Bridge Domain on page 151

Layer 2 Virtual Switches Overview

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

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

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

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

For more information about configuring a routing instance for Layer 2 VPN, see the [Junos OS VPNs Configuration Guide](#). For a detailed Layer 2 VPN example configuration, see the [Junos OS Feature Guides](#).

Related Documentation

- [Configuring a Layer 2 Control Protocol Routing Instance on page 18](#)
- [Layer 2 Learning and Forwarding for Bridge Domains Overview on page 149](#)
- [Layer 2 Protocol Tunneling Through a Network Overview on page 250](#)

Layer 2 Learning and Forwarding for Bridge Domains Overview

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



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

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

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

Related Documentation

- [Layer 2 Learning and Forwarding Overview on page 195](#)

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

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



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

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

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

**Related
Documentation**

- Layer 2 Learning and Forwarding Overview on page 195

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

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

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

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

**Related
Documentation**

- Layer 2 Learning and Forwarding Overview on page 195

Configuring Layer 2 Bridging

- Configuring Bridge Domains for Layer 2 Bridging and Layer 3 IP Routing on page 151
- Configuring Layer 2 Virtual Switches on page 160
- Configuring Layer 2 Learning and Forwarding for Bridge Domains on page 166
- Configuring Layer 2 Learning and Forwarding for Bridge Domains Functioning as Switches with Layer 2 Trunk Ports on page 170
- Configuring Layer 3 Tunnel Services Interfaces on an MX Series Router with a DPC on page 172

Configuring Bridge Domains for Layer 2 Bridging and Layer 3 IP Routing

- Configuring a Bridge Domain on page 151
- Configuring VLAN Identifiers for Bridge Domains and VPLS Routing Instances on page 152
- Configuring Integrated Routing and Bridging for Bridge Domains on page 158
- Configuring Bridge Domains as Switches for Layer 2 Trunk Ports on page 159

Configuring a Bridge Domain

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

To enable a bridge domain, include the following statements:

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

```
}
```

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

For the **vlan-id** statement, you can specify either a valid VLAN identifier or the **none** or **all** options. For information about VLAN identifiers and VLAN tags for a bridge domain, see “Guidelines for Configuring VLAN Identifiers for Bridge Domains and VPLS Routing Instances” on page 150 and “Configuring VLAN Identifiers for Bridge Domains and VPLS Routing Instances” on page 152.

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



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

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

You can also configure spanning tree protocols to prevent forwarding loops. For more information, see “Spanning-Tree Protocols Supported on MX Series Routers” on page 205.

In Junos OS Release 8.5 and later, you can configure IGMP snooping for a bridge domain. For more information, see the [Junos OS Multicast Protocols Configuration Guide](#).

Related Documentation

- Configuring Integrated Routing and Bridging for Bridge Domains on page 158
- Layer 2 Learning and Forwarding Overview on page 195
- Layer 2 Learning and Forwarding for Bridge Domains Overview on page 149
- Layer 2 Learning and Forwarding for Bridge Domains Functioning as Switches with Layer 2 Trunk Ports on page 149

Configuring VLAN Identifiers for Bridge Domains and VPLS Routing Instances

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

- By using the **input-vlan-map** and the **output-vlan-map** statements at the **[edit interfaces interface-name]** or **[edit logical-systems logical-system-name interfaces interface-name]** hierarchy level to configure VLAN mapping. For information about configuring input and output VLAN maps to stack and rewrite VLAN tags in incoming or outgoing frames, see the [Junos OS Network Interfaces Configuration Guide](#).

- By using either the **vlan-id** statement or the **vlan-tags** statement to configure a normalizing VLAN identifier. This topic describes how normalizing VLAN identifiers are processed and translated in a bridge domain or a VPLS routing instance.

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

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



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

To configure a VLAN identifier for a bridge domain, include either the **vlan-id** or the **vlan-tags** statement at the **[edit interfaces *interface-name* unit *logic-unit-number* family bridge]** or **[edit logical-systems *logical-system-name* interfaces *interface-name* unit *logic-unit-number* family bridge]** hierarchy level, and then include that logical interface in the bridge domain configuration. For more information about configuring a bridge domain, see “Configuring a Bridge Domain” on page 151.

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



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

The VLAN tags associated with the inbound logical interface are compared with the normalizing VLAN identifier. If the tags are different, they are rewritten as described in Table 8 on page 156. The source MAC address of a received packet is learned based on the normalizing VLAN identifier.



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

If the VLAN tags associated with the outbound logical interface and the normalizing VLAN identifier are different, the normalizing VLAN identifier is rewritten to match the VLAN tags of the outbound logical interface, as described in Table 9 on page 157.

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

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

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

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

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

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

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

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

```

    }
  }
  interface ge-1/1/2 {
    encapsulation ethernet-vpls;
    unit 0;
  }
  routing-instance {
    V1 {
      instance-type vpls;
      vlan-id all;
      interface ge-1/1/1.1;
      interface ge-1/1/2.0;
    }
  }
}

```



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

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

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

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

1. When a packet is received on a physical port, it is accepted only if the VLAN identifier of the packet matches the VLAN identifier of one of the logical interfaces configured on that port.
2. The VLAN tags of the received packet are then compared with the normalizing VLAN identifier. If the VLAN tags of the packet are different from the normalizing VLAN identifier, the VLAN tags are rewritten as described in Table 8 on page 156.
3. If the source MAC address of the received packet is not present in the source MAC table, it is learned based on the normalizing VLAN identifier.
4. The packet is then forwarded toward one or more outbound Layer 2 logical interfaces based on the destination MAC address. A packet with a known unicast destination MAC address is forwarded only to one outbound logical interface. For each outbound Layer 2 logical interface, the normalizing VLAN identifier configured for the bridge domain or VPLS routing instance is compared with the VLAN tags configured on that logical interface. If the VLAN tags associated with an outbound logical interface do

not match the normalizing VLAN identifier configured for the bridge domain or VPLS routing instance, the VLAN tags are rewritten as described in Table 9 on page 157.

The tables below show how VLAN tags are applied for traffic sent to and from the bridge domain, depending on how the **vlan-id** and **vlan-tags** statements are configured for the bridge domain and on how VLAN identifiers are configured for the logical interfaces in a bridge domain or VPLS routing instance. Depending on your configuration, the following rewrite operations are performed on VLAN tags:

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

Table 8 on page 156 shows specific examples of how the VLAN tags for packets sent to the bridge domain are processed and translated, depending on your configuration. “–” means that the statement is not supported for the specified logical interface VLAN identifier. “No operation” means that the VLAN tags of the received packet are not translated for the specified input logical interface.

Table 8: Statement Usage and Input Rewrite Operations for VLAN Identifiers for a Bridge Domain

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

Table 8: Statement Usage and Input Rewrite Operations for VLAN Identifiers for a Bridge Domain (*continued*)

VLAN Identifier of Logical Interface	VLAN Configurations for Bridge Domain			
	vlan-id none	vlan-id 200	vlan-id all	vlan tags outer 100 inner 300
vlan-tags outer 200 inner-range 10-100	–	–	pop 200	–

Table 9 on page 157 shows specific examples of how the VLAN tags for packets sent from the bridge domain are processed and translated, depending on your configuration. “–” means that the statement is not supported for the specified logical interface VLAN identifier. “No operation” means that the VLAN tags of the outbound packet are not translated for the specified output logical interface.

Table 9: Statement Usage and Output Rewrite Operations for VLAN Identifiers for a Bridge Domain

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

Related Documentation

- Configuring Integrated Routing and Bridging for Bridge Domains on page 158
- Layer 2 Learning and Forwarding Overview on page 195
- Layer 2 Learning and Forwarding for Bridge Domains Overview on page 149
- Layer 2 Learning and Forwarding for Bridge Domains Functioning as Switches with Layer 2 Trunk Ports on page 149

Configuring Integrated Routing and Bridging for Bridge Domains

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



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

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

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

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

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



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

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



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

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



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

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

In Junos OS Release 9.0 and later, IRB interfaces are supported for multicast snooping. For more information about multicast snooping, see the [Junos OS Multicast Protocols Configuration Guide](#).

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

Related Documentation

- Guidelines for Configuring VLAN Identifiers for Bridge Domains and VPLS Routing Instances on page 150
- Configuring VLAN Identifiers for Bridge Domains and VPLS Routing Instances on page 152

Configuring Bridge Domains as Switches for Layer 2 Trunk Ports

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

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

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

```
[edit bridge-domains]
bridge-domain-name {
  vlan-id number;
  vlan-id-list [ vlan-id-numbers ];
  ....
}
```

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

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

**Related
Documentation**

- Configuring a Bridge Domain on page 151

Configuring Layer 2 Virtual Switches

- Configuring a Layer 2 Virtual Switch on page 160
- Configuring a Layer 2 Virtual Switch with a Layer 2 Trunk Port on page 162
- Configuring VPLS Ports in a Virtual Switch on page 163
- Configuring Integrated Routing and Bridging for a Bridge Domain in a Layer 2 Virtual Switch on page 164

Configuring a Layer 2 Virtual Switch

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

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

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



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

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

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

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

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

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

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



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



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

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

For information about how to configure spanning tree protocols, see the [Junos OS Feature Guides](#).

Related Documentation

- Guidelines for Configuring VLAN Identifiers for Bridge Domains and VPLS Routing Instances on page 150
- Configuring VLAN Identifiers for Bridge Domains and VPLS Routing Instances on page 152
- Configuring Integrated Routing and Bridging for a Bridge Domain in a Layer 2 Virtual Switch on page 164

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

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

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

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

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

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



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

Related Documentation

- Layer 2 Learning and Forwarding for Bridge Domains Functioning as Switches with Layer 2 Trunk Ports on page 149

Configuring VPLS Ports in a Virtual Switch

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

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

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

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

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

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

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

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

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

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

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

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

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

Related Documentation

- [Configuring a Bridge Domain on page 151](#)

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

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

information about how to configure a routing interface, see the [Junos OS Network Interfaces Configuration Guide](#).



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

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

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

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

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

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



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

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

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



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

- Related Documentation**
- Guidelines for Configuring VLAN Identifiers for Bridge Domains and VPLS Routing Instances on page 150
 - Configuring VLAN Identifiers for Bridge Domains and VPLS Routing Instances on page 152

Configuring Layer 2 Learning and Forwarding for Bridge Domains

- Disabling MAC Learning for a Bridge Domain or Logical Interface on page 166
- Configuring Static MAC Addresses for Logical Interfaces in a Bridge Domain on page 167
- Configuring the Size of the MAC Address Table on page 168
- Limiting MAC Addresses Learned from an Interface in a Bridge Domain on page 168
- Enabling MAC Accounting for a Bridge Domain on page 170

Disabling MAC Learning for a Bridge Domain or Logical Interface

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

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

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

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

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



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



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

Related Documentation

- Configuring Static MAC Addresses for Logical Interfaces in a Bridge Domain on page 167
- Configuring the Size of the MAC Address Table on page 168
- Limiting MAC Addresses Learned from an Interface in a Bridge Domain on page 168
- Enabling MAC Accounting for a Bridge Domain on page 170

Configuring Static MAC Addresses for Logical Interfaces in a Bridge Domain

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

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

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

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



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

- Related Documentation**
- Disabling MAC Learning for a Bridge Domain or Logical Interface on page 166
 - Configuring the Size of the MAC Address Table on page 168
 - Limiting MAC Addresses Learned from an Interface in a Bridge Domain on page 168
 - Enabling MAC Accounting for a Bridge Domain on page 170

Configuring the Size of the MAC Address Table

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

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

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

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

- Related Documentation**
- Disabling MAC Learning for a Bridge Domain or Logical Interface on page 166
 - Configuring Static MAC Addresses for Logical Interfaces in a Bridge Domain on page 167
 - Limiting MAC Addresses Learned from an Interface in a Bridge Domain on page 168
 - Enabling MAC Accounting for a Bridge Domain on page 170

Limiting MAC Addresses Learned from an Interface in a Bridge Domain

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

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

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

```

        interface-mac-limit limit;
    }
}

```

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

```

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

```

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

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

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

```

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

```

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

```

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

```

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

- Related Documentation**
- Disabling MAC Learning for a Bridge Domain or Logical Interface on page 166
 - Configuring Static MAC Addresses for Logical Interfaces in a Bridge Domain on page 167
 - Configuring the Size of the MAC Address Table on page 168
 - Enabling MAC Accounting for a Bridge Domain on page 170

Enabling MAC Accounting for a Bridge Domain

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

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

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

- Related Documentation**
- Disabling MAC Learning for a Bridge Domain or Logical Interface on page 166
 - Configuring Static MAC Addresses for Logical Interfaces in a Bridge Domain on page 167
 - Configuring the Size of the MAC Address Table on page 168
 - Limiting MAC Addresses Learned from an Interface in a Bridge Domain on page 168

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

- Disabling MAC Learning for a Set of Bridge Domains on page 170
- Limiting the Number of MAC Addresses Learned from a Trunk Port on page 171
- Configuring the Size of the MAC Address Table for a Set of Bridge Domains on page 172
- Enabling MAC Accounting for a Set of Bridge Domains on page 172

Disabling MAC Learning for a Set of Bridge Domains

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

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

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

- Related Documentation**
- Limiting the Number of MAC Addresses Learned from a Trunk Port on page 171
 - Configuring the Size of the MAC Address Table for a Set of Bridge Domains on page 172

- Enabling MAC Accounting for a Set of Bridge Domains on page 172

Limiting the Number of MAC Addresses Learned from a Trunk Port

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

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

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

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

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

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

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

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

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

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

Related Documentation

- Disabling MAC Learning for a Set of Bridge Domains on page 170
- Configuring the Size of the MAC Address Table for a Set of Bridge Domains on page 172
- Enabling MAC Accounting for a Set of Bridge Domains on page 172

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

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

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

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

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

Related Documentation

- Disabling MAC Learning for a Set of Bridge Domains on page 170
- Limiting the Number of MAC Addresses Learned from a Trunk Port on page 171
- Enabling MAC Accounting for a Set of Bridge Domains on page 172

Enabling MAC Accounting for a Set of Bridge Domains

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

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

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

Related Documentation

- Disabling MAC Learning for a Set of Bridge Domains on page 170
- Limiting the Number of MAC Addresses Learned from a Trunk Port on page 171
- Configuring the Size of the MAC Address Table for a Set of Bridge Domains on page 172

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

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

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

```
[edit chassis]
fpc slot-number {
  pic number {
    tunnel-services {
      bandwidth (1g | 10g);
    }
  }
}
```

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

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

You can also specify the amount of bandwidth to allocate for tunnel traffic on each Packet Forwarding Engine by including the **bandwidth (1g | 10g)** statement at the **[edit chassis fpc slot-number pic pic-number]** hierarchy level:

- **1g** indicates that 1 Gbps of bandwidth is reserved for tunnel traffic. Configure this option only for a Packet Forwarding Engine on a Gigabit Ethernet 40-port DPC.
- **10g** indicates that 10 Gbps of bandwidth is reserved for tunnel traffic. Configure this option only for a Packet Forwarding Engine on a 10-Gigabit Ethernet 4-port DPC.

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

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

To verify that the tunnel interfaces have been created, issue the **show interfaces terse** operational mode command. For more information, see the [Junos OS Interfaces Command Reference](#).

For additional information about tunnel services, see the “Tunnel Services” chapter in the [Junos OS Services Interfaces Configuration Guide](#).

Related Documentation

- MX Series Router Architecture on page 3
- MX Series Router Packet Forwarding and Data Flow on page 5


- [Line Cards Supported on MX Series Routers](#) on page 5

CHAPTER 13


Summary of Layer 2 Bridging Configuration Statements

The following sections explain each of the bridge domain configuration statements. The statements are organized alphabetically.

bandwidth

Syntax	bandwidth (1g 10g);
Hierarchy Level	[edit chassis fpc slot-number pic number tunnel-services]
Release Information	Statement introduced in Junos OS Release 8.2.
Description	On MX Series routers only, specify the amount of bandwidth to reserve for tunnel services.
Options	<p>1g—Specify a bandwidth of 1 Gbps on the Packet Forwarding Engine connected to a Gigabit Ethernet 40-port Dense Port Concentrator (DPC).</p> <p>10g—Specify a bandwidth of 10 Gbps on the Packet Forwarding Engine connected to 10-Gigabit Ethernet 4-port DPC.</p>
	<div> NOTE: If you specify a bandwidth that is not compatible with the type of DPC and Packet Forwarding Engine, tunnel services are not activated. For example, you cannot specify a bandwidth of 1 Gbps for a Packet Forwarding Engine on a 10-Gigabit Ethernet 4-port DPC.</div>
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">Configuring Layer 3 Tunnel Services Interfaces on an MX Series Router with a DPC on page 172

bridge-domains

Syntax	<pre> bridge-domains { bridge-domain-name { bridge-options { ...bridge-options-configuration... } domain-type bridge; interface interface-name; no-irb-layer-2-copy; routing-interface routing-interface-name; vlan-id (all none number); vlan-id-list [vlan-id-numbers]; vlan-tags outer number inner number; bridge-options { interface interface-name { static-mac mac-address; } interface-mac-limit limit; mac-statistics; mac-table-size limit; no-mac-learning; } } } </pre>
Hierarchy Level	[edit], [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. Support for logical systems added in Junos OS Release 9.6. Support for the no-irb-layer-2-copy statement added in Junos OS Release 10.2.
Description	(MX Series routers only) Configure a domain that includes a set of logical ports that share the same flooding or broadcast characteristics in order to perform Layer 2 bridging.
Options	bridge-domain-name —Name of the bridge domain.
<div style="display: flex; align-items: center;">  <div> <p>NOTE: You cannot use the slash (/) character as part of the bridge domain name. If you do, the configuration will not commit.</p> </div> </div>	
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 a Bridge Domain on page 151 Configuring a Layer 2 Virtual Switch on page 160

- Example: Configuring E-LINE and E-LAN Services for a PBB Network on MX Series Routers

bridge-options

Syntax	<pre>bridge-options { interface <i>interface-name</i>; static-mac <i>static-mac-address</i>; } interface-mac-limit <i>limit</i>; packet-action drop; } mac-statistics; mac-table-size <i>limit</i>; no-mac-learning; }</pre>
Hierarchy Level	<pre>[edit bridge-domains <i>bridge-domain-name</i>], [edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> bridge-domains <i>bridge-domain-name</i>], [edit routing-instances <i>routing-instance-name</i> bridge-domains <i>bridge-domain-name</i>]</pre>
Release Information	<p>Statement introduced in Junos OS Release 8.4.</p> <p>Support for logical systems added in Junos OS Release 9.6.</p>
Description	<p>(MX Series routers only) Configure Layer 2 learning and forwarding properties for a bridge domain 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>
Related Documentation	<ul style="list-style-type: none"> • Layer 2 Learning and Forwarding for Bridge Domains Overview on page 149

domain-type

Syntax	domain-type bridge;
Hierarchy Level	[edit bridge-domains <i>bridge-domain-name</i>], [edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> bridge-domains <i>bridge-domain-name</i>], [edit routing-instances <i>routing-instance-name</i> bridge-domains <i>bridge-domain-name</i>]
Release Information	Statement introduced in Junos OS Release 8.4. Support for logical systems added in Junos OS Release 9.6.
Description	(MX Series routers only) Define the type of domain for a Layer 2 bridge domain.
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 a Bridge Domain on page 151Configuring a Layer 2 Virtual Switch on page 160

interface

Syntax	interface <i>interface-name</i> ;
Hierarchy Level	[edit bridge-domains <i>bridge-domain-name</i>], [edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> bridge-domains <i>bridge-domain-name</i>], [edit routing-instances <i>routing-instance-name</i> bridge-domains <i>bridge-domain-name</i>], [edit bridge-domains <i>bridge-domain-name</i>],
Release Information	Statement introduced in Junos OS Release 8.4. 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.
Description	(MX Series routers only) Specify the logical interfaces to include in the bridge domain, VPLS instance, or virtual switch.
Options	<i>interface-name</i> —Name of a logical interface. For more information about how to configure logical interfaces, see the Junos OS Network Interfaces Configuration Guide .
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 a Bridge Domain on page 151Configuring a Layer 2 Virtual Switch on page 160

interface-mac-limit

Syntax	<pre>interface-mac-limit <i>limit</i> { packet-action drop; }</pre>
Hierarchy Level	<pre>[edit bridge-domains <i>bridge-domain-name</i> bridge-options], [edit bridge-domains <i>bridge-domain-name</i> bridge-options interface <i>interface-name</i>], [edit logical-systems <i>logical-system-name</i> bridge-domains <i>bridge-domain-name</i> bridge-options], [edit logical-systems <i>logical-system-name</i> bridge-domains <i>bridge-domain-name</i> bridge-options interface <i>interface-name</i>], [edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> bridge-domains <i>bridge-domain-name</i> bridge-options], [edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> bridge-domains <i>bridge-domain-name</i> bridge-options interface <i>interface-name</i>], [edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> switch-options], [edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> switch-options interface <i>interface-name</i>], [edit logical-systems <i>logical-system-name</i> switch-options], [edit logical-systems <i>logical-system-name</i> switch-options interface <i>interface-name</i>], [edit routing-instances <i>routing-instance-name</i> bridge-domains <i>bridge-domain-name</i> bridge-options], [edit routing-instances <i>routing-instance-name</i> bridge-domains <i>bridge-domain-name</i> bridge-options interface <i>interface-name</i>], [edit routing-instances <i>routing-instance-name</i> switch-options], [edit routing-instances <i>routing-instance-name</i> switch-options interface <i>interface-name</i>], [edit switch-options], [edit switch-options interface <i>interface-name</i>]</pre>
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>
Description	<p>(MX Series routers only) Configure a limit to the number of MAC addresses that can be learned from a bridge domain, virtual switch, or set of bridge domains.</p>
Default	1024 MAC addresses for each logical interface.
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 on page 149• Layer 2 Learning and Forwarding for Bridge Domains Functioning as Switches with Layer 2 Trunk Ports on page 149 |
|------------------------------|---|

mac-statistics

Syntax	mac-statistics;
Hierarchy Level	[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]
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.
Description	(MX Series routers only) Enable MAC accounting either for a specific bridge domain, or for a set of bridge domains associated with a Layer 2 trunk port.
Default	disabled
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">• Layer 2 Learning and Forwarding for Bridge Domains Overview on page 149• Layer 2 Learning and Forwarding for Bridge Domains Functioning as Switches with Layer 2 Trunk Ports on page 149

mac-table-size

Syntax	<pre>mac-table-size <i>limit</i> { packet-action drop; }</pre>
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]</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>
Description	<p>Modify the size of the MAC address table for the bridge domain, a set of bridge domains associated with a trunk port, or a virtual switch. The default is 5120 MAC addresses.</p>
Options	<p>limit—Specify the maximum number of addresses in the MAC address table.</p> <p>Range: 16 through 1,048,575 MAC addresses</p> <p>Default: 5120 MAC addresses</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 on page 149 Layer 2 Learning and Forwarding for Bridge Domains Functioning as Switches with Layer 2 Trunk Ports on page 149

no-irb-layer-2-copy

Syntax	no-irb-layer-2-copy;
Hierarchy Level	[edit bridge-domains], [edit logical-routers <i>logical-router-name</i> bridge-domains], [edit routing-instances <i>routing-instance-name</i> bridge-domains]
Release Information	Statement introduced in Junos OS Release 10.2.
Description	If you include this statement when using port mirroring with Integrated Routing and Bridging (IRB), then the packet is mirrored as a Layer 3 packet. By default, the packet is mirrored as a Layer 2 packet. This statement is also supported if a routing instance is set to type VPLS.
Usage Guidelines	See "Configuring a Bridge Domain" on page 151
Required Privilege Level	view-level—To view this statement in the configuration. control-level—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">Configuring a Layer 2 Virtual Switch on page 160

no-mac-learning


Syntax	no-mac-learning;
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> bridge-domains <i>bridge-domain-name</i> bridge-options interface <i>interface-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> 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 interface <i>interface-name</i>],</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 bridge-domains <i>bridge-domain-name</i> bridge-options interface <i>interface-name</i>],</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> bridge-domains <i>bridge-domain-name</i> bridge-options interface <i>interface-name</i>],</p> <p>[edit routing-instances <i>routing-instance-name</i> switch-options],</p> <p>[edit 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 bridge domain configured within a virtual switch.</p> <p>Support for logical systems added in Junos OS Release 9.6.</p>
Description	(MX Series routers only) Disable MAC learning for a virtual switch, for a bridge domain, for a specific logical interface in a bridge domain, or for a set of bridge domains associated with a Layer 2 trunk port.
Default	MAC learning is enabled. Use no-mac-learning to disable MAC learning.
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 on page 149 Layer 2 Learning and Forwarding for Bridge Domains Functioning as Switches with Layer 2 Trunk Ports on page 149

packet-action

Syntax	packet-action drop;
Hierarchy Level	<p>[edit bridge-domains <i>bridge-domain-name</i> bridge-options interface <i>interface-name</i> interface-mac-limit <i>limit</i>],</p> <p>[edit bridge-domains <i>bridge-domain-name</i> bridge-options interface-mac-limit <i>limit</i>],</p> <p>[edit logical-systems <i>logical-system-name</i> bridge-domains <i>bridge-domain-name</i> bridge-options interface <i>interface-name</i> interface-mac-limit <i>limit</i>],</p> <p>[edit logical-systems <i>logical-system-name</i> bridge-domains <i>bridge-domain-name</i> bridge-options interface-mac-limit <i>limit</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> bridge-options interface <i>interface-name</i> interface-mac-limit <i>limit</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> bridge-options interface-mac-limit <i>limit</i>],</p> <p>[edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> switch-options interface <i>interface-name</i> interface-mac-limit <i>limit</i>],</p> <p>[edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> switch-options interface-mac-limit <i>limit</i>],</p> <p>[edit logical-systems <i>logical-system-name</i> switch-options interface-mac-limit <i>limit</i>],</p> <p>[edit protocols l2-learning global-mac-limit <i>limit</i>],</p> <p>[edit routing-instances <i>routing-instance-name</i> bridge-domains <i>bridge-domain-name</i> bridge-options interface <i>interface-name</i> interface-mac-limit <i>limit</i>],</p> <p>[edit routing-instances <i>routing-instance-name</i> bridge-domains <i>bridge-domain-name</i> bridge-options interface-mac-limit <i>limit</i>],</p> <p>[edit routing-instances <i>routing-instance-name</i> switch-options interface <i>interface-name</i> interface-mac-limit <i>limit</i>],</p> <p>[edit routing-instances <i>routing-instance-name</i> switch-options interface-mac-limit <i>limit</i>],</p> <p>[edit switch-options interface-mac-limit <i>limit</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>
Description	(MX Series routers only) Specify that packets for new source MAC addresses be dropped after the MAC address limit is reached. If this statement is not configured, then packets for new source MAC addresses are forwarded by default.
Default	Disabled. The default is for packets for new source MAC addresses to be forwarded after the MAC address limit is reached.
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 on page 149

- Layer 2 Learning and Forwarding for Bridge Domains Functioning as Switches with Layer 2 Trunk Ports on page 149

routing-interface

Syntax	<code>routing-interface <i>routing-interface-name</i>;</code>
Hierarchy Level	[edit bridge-domains <i>bridge-domain-name</i>], [edit logical-systems <i>logical-system-name</i> bridge-domains <i>bridge-domain-name</i>], [edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> bridge-domains <i>bridge-domain-name</i>], [edit routing-instances <i>routing-instance-name</i> bridge-domains <i>bridge-domain-name</i>]
Release Information	Statement introduced in Junos OS Release 8.4. Support for logical systems added in Junos OS Release 9.6.
Description	(MX Series routers only) Specify a routing interface to include in a bridge domain or a VPLS routing instance.
Options	<i>routing-interface-name</i> —Name of the routing interface to include in the bridge domain or the VPLS routing instance. The format of the routing interface name is irb.x , where <i>x</i> is the unit number of the routing interface you configured at the [edit interfaces irb] hierarchy level. For more information about how to configure a routing interface, see the Junos OS Network Interfaces Configuration Guide .
<div>  <p>NOTE: You can specify only one routing interface for each bridge domain or VPLS instance.</p> </div>	
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 a Bridge Domain on page 151 • Configuring a Layer 2 Virtual Switch on page 160

static-mac

Syntax	<code>static-mac <i>mac-address</i> { vlan-id <i>number</i>; }</code>
Hierarchy Level	[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 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 interface <i>interface-name</i>], [edit routing-instances <i>routing-instance-name</i> bridge-domains <i>bridge-domain-name</i> bridge-options interface <i>interface-name</i>]
Release Information	Statement introduced in Junos OS Release 8.4. Support for logical systems added in Junos OS Release 9.6.
Description	(MX Series routers only) Configure a static MAC address for a logical interface in a bridge domain.
Options	<i>mac-address</i> —MAC address <i>vlan-id number</i> —(Optional) VLAN identifier to associate with static 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">Layer 2 Learning and Forwarding for Bridge Domains Overview on page 149

switch-options

Syntax	<pre>switch-options { interface <i>interface-name</i> { interface-mac-limit <i>limit</i>; } interface-mac-limit <i>limit</i> { packet-action drop; } mac-statistics; mac-table-size <i>limit</i> { packet-action drop; } no-mac-learning; }</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 9.2. Support for logical systems added in Junos OS Release 9.6.
Description	Configure Layer 2 learning and forwarding properties for a set of bridge domains.
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"> Layer 2 Learning and Forwarding for Bridge Domains Functioning as Switches with Layer 2 Trunk Ports on page 149

tunnel-services

Syntax	tunnel-services { bandwidth (1g 10g); }
Hierarchy Level	[edit chassis fpc <i>slot-number</i> pic <i>number</i>]
Release Information	Statement introduced in Junos OS Release 8.2.
Description	For MX Series routers, configure the amount of bandwidth for tunnel services.
Options	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	<ul style="list-style-type: none">Configuring Layer 3 Tunnel Services Interfaces on an MX Series Router with a DPC on page 172

vlan-id (Bridge Domain)

Syntax	<code>vlan-id (all none <i>number</i>);</code>
Hierarchy Level	<code>[edit bridge-domains <i>bridge-domain-name</i>],</code> <code>[edit logical-systems <i>logical-system-name</i> bridge-domainss <i>bridge-domain-name</i>],</code> <code>[edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i></code> <code> bridge-domains <i>bridge-domain-name</i>],</code> <code>[edit routing-instances <i>routing-instance-name</i> bridge-domains <i>bridge-domain-name</i>]</code>
Release Information	Statement introduced in Junos OS Release 8.4. Support for Layer 2 trunk ports added in Junos OS Release 9.2. Support for logical systems added in Junos OS Release 9.6.
Description	(MX Series routers only) Specify a VLAN identifier (VID) to include in the packets sent to and from the bridge domain or a VPLS routing instance.



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

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



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

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



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

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

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

- Related Documentation**
- Configuring a Bridge Domain on page 151
 - Configuring VLAN Identifiers for Bridge Domains and VPLS Routing Instances on page 152
 - Configuring Bridge Domains as Switches for Layer 2 Trunk Ports on page 159
 - Configuring a Layer 2 Virtual Switch on page 160
 - Example: Configuring E-LINE and E-LAN Services for a PBB Network on MX Series Routers

vlan-id-list

Syntax	<code>vlan-id-list [<i>vlan-id-numbers</i>];</code>
Hierarchy Level	<code>[edit bridge-domains <i>bridge-domain-name</i>],</code> <code>[edit logical-systems <i>logical-system-name</i> bridge-domains <i>bridge-domain-name</i>],</code> <code>[edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i></code> <code>bridge-domains <i>bridge-domain-name</i>],</code> <code>[edit routing-instances <i>routing-instance-name</i> bridge-domains <i>bridge-domain-name</i>]</code>
Release Information	Statement introduced in Junos OS Release 9.4. Support for logical systems added in Junos OS Release 9.6.
Description	(MX Series routers only) Specify a VLAN identifier list to use for a bridge domain in trunk mode.
Options	<i>vlan-id-numbers</i> —Valid VLAN identifiers. You can combine individual numbers with range lists including a hyphen. Range: 0 through 4095
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 a Bridge Domain on page 151• Configuring VLAN Identifiers for Bridge Domains and VPLS Routing Instances on page 152

vlan-tags

Syntax	<code>vlan-tags outer <i>number</i> inner <i>number</i>;</code>
Hierarchy Level	<code>[edit bridge-domains <i>bridge-domain-name</i>],</code> <code>[edit logical-systems <i>logical-system-name</i> bridge-domains <i>bridge-domain-name</i>],</code> <code>[edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i></code> <code> bridge-domains <i>bridge-domain-name</i>],</code> <code>[edit routing-instances <i>routing-instance-name</i> bridge-domains <i>bridge-domain-name</i>]</code>
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>
Description	(MX Series routers only) Specify dual VLAN identifier tags for a bridge domain or a 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"> Configuring a Bridge Domain on page 151 Configuring VLAN Identifiers for Bridge Domains and VPLS Routing Instances on page 152 Configuring a Layer 2 Virtual Switch on page 160.

PART 5

Layer 2 Address Learning and Forwarding

- Layer 2 Learning and Forwarding on page 195
- Summary of Layer 2 Address Learning and Forwarding Configuration Statements on page 199

Layer 2 Learning and Forwarding

- Layer 2 Learning and Forwarding Overview on page 195
- Configuring Layer 2 Learning and Forwarding on page 196
- Disabling Layer 2 Learning and Forwarding on page 198

Layer 2 Learning and Forwarding Overview

On MX Series routers only, you can configure Layer 2 MAC address and VLAN learning and forwarding properties in support of Layer 2 bridging. The router learns unicast media access control (MAC) addresses to avoid flooding the packets to all the ports in a bridge domain. The MX Series router creates a source MAC entry in its source and destination MAC tables for each MAC address learned from packets received on ports that belong to the bridge domain.

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

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

For more information about how to configure bridge domains and virtual switches, see “Configuring a Bridge Domain” on page 151 and “Configuring a Layer 2 Virtual Switch” on page 160.

Related Documentation

- Configuring the MAC Table Timeout Interval on page 196
- Enabling MAC Accounting on page 196
- Limiting the Number of MAC Addresses Learned from Each Logical Interface on page 197
- Disabling Layer 2 Learning and Forwarding on page 198

Configuring Layer 2 Learning and Forwarding

- Configuring the MAC Table Timeout Interval on page 196
- Enabling MAC Accounting on page 196
- Limiting the Number of MAC Addresses Learned from Each Logical Interface on page 197

Configuring the MAC Table Timeout Interval

By default, the timeout interval for all entries in the MAC table is 300 seconds. You can modify the timeout interval for MAC table entries on an MX Series router. You cannot modify the timeout interval only for specific MAC table entries, such as for a bridge domain or a virtual switch.



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

To modify the timeout interval for the MAC table for the entire routing platform, include the **global-mac-table-aging-time seconds** statement at the **[edit protocols l2-learning]** hierarchy level:

```
[edit protocols l2-learning]
global-mac-table-aging-time seconds;
```

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

Related Documentation

- Layer 2 Learning and Forwarding Overview on page 195
- Enabling MAC Accounting on page 196
- Limiting the Number of MAC Addresses Learned from Each Logical Interface on page 197
- Disabling Layer 2 Learning and Forwarding on page 198

Enabling MAC Accounting

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

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

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

Related Documentation

- Layer 2 Learning and Forwarding Overview on page 195
- Configuring the MAC Table Timeout Interval on page 196
- Limiting the Number of MAC Addresses Learned from Each Logical Interface on page 197

- Disabling Layer 2 Learning and Forwarding on page 198

Limiting the Number of MAC Addresses Learned from Each Logical Interface

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

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

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

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

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

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

You can also configure a limit to the number of MAC address learned from all the interfaces in a bridge domain or from a specific logical interface only. For more information, see “Layer 2 Learning and Forwarding for Bridge Domains Overview” on page 149.



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

Related Documentation

- Layer 2 Learning and Forwarding Overview on page 195
- Configuring the MAC Table Timeout Interval on page 196
- Enabling MAC Accounting on page 196
- Disabling Layer 2 Learning and Forwarding on page 198

Disabling Layer 2 Learning and Forwarding

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

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

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

For information about how to configure a virtual switch, see “Configuring a Layer 2 Virtual Switch” on page 160.

Related Documentation

- Layer 2 Learning and Forwarding Overview on page 195
- Configuring the MAC Table Timeout Interval on page 196
- Enabling MAC Accounting on page 196
- Limiting the Number of MAC Addresses Learned from Each Logical Interface on page 197

CHAPTER 15

Summary of Layer 2 Address Learning and Forwarding Configuration Statements

The following sections explain each of the Layer 2 address learning and forwarding configuration statements. These statements are organized alphabetically.

global-mac-limit

Syntax	<code>global-mac-limit <i>limit</i> { packet-action drop; }</code>
Hierarchy Level	[edit protocols l2-learning]
Release Information	Statement introduced in Junos OS Release 8.4. Support for logical systems added in Junos OS Release 9.6.
Description	(MX Series routers only) Limit the number of media access control (MAC) addresses learned from the logical interfaces on the router.
Default	393,215 MAC addresses
Options	<i>limit</i> —Number of MAC addresses that can be learned systemwide. Range: 20 through 1,048,575 The remaining statement is explained separately in the “Summary of Bridge Domain Configuration Statements” chapter.
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">Limiting the Number of MAC Addresses Learned from Each Logical Interface on page 197

global-mac-statistics

Syntax	global-mac-statistics;
Hierarchy Level	[edit protocols l2-learning]
Release Information	Statement introduced in Junos OS Release 9.2. Support for logical systems added in Junos OS Release 9.6.
Description	(MX Series routers only) Enable MAC accounting for the entire router,
Default	disabled
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">Enabling MAC Accounting on page 196

global-mac-table-aging-time

Syntax	global-mac-table-aging-time <i>seconds</i> ;
Hierarchy Level	[edit protocols l2-learning]
Release Information	Statement introduced in Junos OS Release 9.2. Support for logical systems added in Junos OS Release 9.6.
Description	(MX Series routers only) Configure the timeout interval for entries in the MAC table.
Default	300 seconds
Options	<i>seconds</i> —Time elapsed before MAC table entries are timed out and entries are deleted from the table. Range: 10 through 1 million
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 the MAC Table Timeout Interval on page 196

global-no-mac-learning

Syntax	global-no-mac-learning;
Hierarchy Level	[edit protocols l2-learning]
Release Information	Statement introduced in Junos OS Release 9.2. Support for logical systems added in Junos OS Release 9.6.
Description	(MX Series routers only) Disable MAC learning for the entire router.
Default	MAC learning is enabled.
Required Privilege Level	routing—To view this statement in the configuration. routing-control—To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">Disabling Layer 2 Learning and Forwarding on page 198

l2-learning

Syntax	<pre>l2-learning { global-mac-limit <i>limit</i>; global-mac-statistics; global-mac-table-aging-time <i>seconds</i>; global-no-mac-learning; }</pre>
Hierarchy Level	[edit protocols]
Release Information	Statement introduced in Junos OS Release 8.4.
Description	(MX Series routers only) Configure Layer 2 address learning and forwarding properties globally. 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">Layer 2 Learning and Forwarding Overview on page 195

PART 6

Spanning-Tree Protocols

- [Spanning-Tree Protocols Overview on page 205](#)
- [Guidelines for Configuring Spanning-Tree Protocols on page 209](#)
- [Configuring Spanning-Tree Protocols on page 217](#)
- [Spanning-Tree Protocol Options on page 233](#)
- [Examples of Spanning-Tree Protocol Configurations on page 255](#)
- [Summary of Spanning-Tree Protocol Configuration Statements on page 259](#)

Spanning-Tree Protocols Overview

- Spanning-Tree Protocols Supported on MX Series Routers on page 205
- BPDU Overview on page 206

Spanning-Tree Protocols Supported on MX Series Routers

On MX Series routers in a Layer 2 environment, you can configure various spanning-tree protocol versions to create a loop-free topology in Layer 2 networks.

A spanning-tree protocol is a Layer 2 control protocol (L2CP) that calculates the best path through a switched network containing redundant paths. A spanning-tree protocol uses bridge protocol data unit (BPDU) data frames to exchange information with other switches. A spanning-tree protocol 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. The resulting tree topology provides a single active Layer 2 data path between any two end stations.



NOTE: In discussions of spanning-tree protocols, the terms *bridge* and *switch* are often used interchangeably.

The Juniper Networks MX Series 3D Universal Edge Routers support STP, RSTP, MSTP, and VSTP.

- The original Spanning-Tree Protocol (STP) is defined in the IEEE 802.1D 1998 specification. A newer version called Rapid Spanning-Tree Protocol (RSTP) was originally defined in the IEEE 802.1w draft specification and later incorporated into the IEEE 802.1D-2004 specification. A recent version called Multiple Spanning-Tree Protocol (MSTP) was originally defined in the IEEE 802.1s draft specification and later incorporated into the IEEE 802.1Q-2003 specification. The VLAN Spanning-Tree Protocol (VSTP) is compatible with the Per-VLAN Spanning Tree Plus (PVST+) and Rapid-PVST+ protocols supported on Cisco Systems routers and switches.
- RSTP provides faster reconvergence time than the original STP by identifying certain links as point to point and by using protocol handshake messages rather than fixed timeouts. When a point-to-point link fails, the alternate link can transition to the forwarding state without waiting for any protocol timers to expire.

- MSTP provides the capability to logically divide a Layer 2 network into regions. Every region has a unique identifier and can contain multiple instances of spanning trees. All regions are bound together using a Common Instance Spanning Tree (CIST), which is responsible for creating a loop-free topology *across* regions, whereas the Multiple Spanning-Tree Instance (MSTI) controls topology *within* regions. MSTP uses RSTP as a converging algorithm and is fully interoperable with earlier versions of STP.
- VSTP maintains a separate spanning-tree instance for each VLAN. Different VLANs can use different spanning-tree paths. When different VLANs use different spanning-tree paths, the CPU processing resources being consumed increase as more VLANs are configured. VSTP BPDU packets are tagged with the corresponding VLAN identifier and are transmitted to the multicast destination media access control (MAC) address **01-00-0c-cc-cc-cd** with a protocol type of **0x010b**. VSTP BPDUs are tunneled by pure IEEE 802.1q bridges.



NOTE: All virtual switch routing instances configured on an MX Series router are supported using only one spanning-tree process. The Layer 2 control protocol process is named `l2cpd`.

For more information about the various versions of spanning-tree protocols, see the appropriate IEEE specification.

Related Documentation

- Loop Protection for Spanning-Tree Instance Interfaces Overview on page 233
- Root Protection for Spanning-Tree Instance Interfaces Overview on page 236
- BPDU Protection for Spanning-Tree Instance Interfaces Overview on page 238
- VPLS Multihomed Layer 2 Ring and MPLS Infrastructure Overview on page 242

BPDU Overview

In a Layer 2 bridge environment, spanning-tree protocols use data frames called Bridge Protocol Data Units (BPDUs) to exchange information among bridges.

Spanning-tree protocols on peer systems exchange BPDUs, which contain information about port roles, bridge IDs, and root path costs. On each MX Series router, the spanning-tree protocol uses this information 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. The resulting tree topology provides a single active Layer 2 data path between any two end stations.



NOTE: In discussions of spanning-tree protocols, the terms *bridge* and *switch* are often used interchangeably.

The transmission of BPDUs is controlled by the Layer 2 Control Protocol process (`l2cpd`) on MX Series 3D Universal Edge Routers.

The transmission of periodic packets on behalf of the l2cpd process is carried out by periodic packet management (PPM), which, by default, is configured to run on the Packet Forwarding Engine. The ppm process on the Packet Forwarding Engine ensures that the BPDUs are transmitted even when the l2cpd process control plane is unavailable, and keeps the remote adjacencies alive during a unified in-service software upgrade (unified ISSU). However, if you want the distributed PPM (ppmd) process to run on the Routing Engine instead of the Packet Forwarding Engine, you can disable the ppm process on the Packet Forwarding Engine. For more information, see the [Junos OS High Availability Configuration Guide](#).

On MX Series routers with redundant Routing Engines (two Routing Engines that are installed in the same router), you can configure nonstop bridging. Nonstop bridging enables the router to switch from a primary Routing Engine to a backup Routing Engine without losing Layer 2 Control Protocol (L2CP) information. Nonstop bridging uses the same infrastructure as graceful Routing Engine switchover (GRES) to preserve interface and kernel information. However, nonstop bridging also saves L2CP information by running the l2cpd process on the backup Routing Engine.



NOTE: To use nonstop bridging, you must first enable GRES.

Nonstop bridging is supported for the following Layer 2 control protocols:

- Spanning-Tree Protocol (STP)
- Rapid Spanning-Tree Protocol (RSTP)
- Multiple Spanning-Tree Protocol (MSTP)

For more information about GRES and nonstop bridging, see the [Junos OS High Availability Configuration Guide](#).

**Related
Documentation**

- Spanning-Tree Protocols Supported on MX Series Routers on page 205
- Loop Protection for Spanning-Tree Instance Interfaces Overview on page 233
- Root Protection for Spanning-Tree Instance Interfaces Overview on page 236
- BPDU Protection for Spanning-Tree Instance Interfaces Overview on page 238

Guidelines for Configuring Spanning-Tree Protocols

- Spanning-Tree Protocols in Logical Systems on page 209
- IEEE 802.1D STP Version Forced for RSTP or VSTP on page 209
- Basic Configuration of Bridges in Spanning-Tree Instances on page 211
- Physical Interface Configuration for Spanning-Tree Protocol Instances on page 213
- Spanning-Tree Protocol Trace Options on page 216

Spanning-Tree Protocols in Logical Systems

On MX Series routers only, you can configure spanning-tree protocols in logical systems for bridge domains and other virtual-switch routing instances.

When configuring spanning-tree protocols in logical systems for bridge domains and other virtual-switch routing instances, the following guidelines apply:

- You can only configure 16 logical systems.
- Logging is performed for the entire device and not per logical system.
- You cannot restart Layer 2 learning for an individual logical system.

Related Documentation

- Spanning-Tree Protocols Supported on MX Series Routers on page 205

IEEE 802.1D STP Version Forced for RSTP or VSTP

The following topics describe guidelines for configuring Rapid Spanning-Tree Protocol (RSTP) or VLAN Spanning-Tree Protocol (VSTP) to run as the original IEEE 802.1D Spanning-Tree Protocol (STP) version for compatibility with older bridges that do not support RSTP or VSTP:

- RSTP or VSTP Forced to Run as IEEE 802.1D STP on page 210
- Reverting RSTP or VSTP Back From Forced IEEE 802.1D STP on page 210

RSTP or VSTP Forced to Run as IEEE 802.1D STP

On MX Series routers in a Layer 2 environment, you can force the configured Rapid Spanning-Tree Protocol (RSTP) or VLAN Spanning-Tree Protocol (VSTP) to run as the original IEEE 802.1D Spanning-Tree Protocol (STP) version. Configure this option for compatibility with older bridges that do not support RSTP or VSTP.

Keep the following limitations in mind when RSTP or VSTP are run as the original STP version:

- If you configure an instance interface as an edge port, the configuration statement is ignored.
- If you configure point-to-point link mode for an instance interface, the configuration statement is ignored.

Related Documentation

- Spanning-Tree Protocols Supported on MX Series Routers on page 205
- Configuring Rapid Spanning-Tree Protocol on page 217
- Configuring VLAN Spanning-Tree Protocol on page 226
- Reverting RSTP or VSTP Back From Forced IEEE 802.1D STP on page 210
- **force-version** on page 269

Reverting RSTP or VSTP Back From Forced IEEE 802.1D STP

On MX Series routers on which Rapid Spanning-Tree Protocol (RSTP) or VLAN Spanning-Tree Protocol (VSTP) has been forced to run as the original IEEE 802.1D Spanning-Tree Protocol (STP) version, you can revert back to RSTP or VSTP.

To revert from the forced instance of the original IEEE 802.1D STP version back to the configured RSTP or VSTP version:

1. Remove the **force-version** statement from the RSTP or VSTP configuration:

```
user@host# delete force-version
```

Include this statement under the RSTP or VSTP hierarchy level:

- [edit protocols rstp]
- [edit protocols vstp]
- [edit routing-instances *routing-instance-name* protocols rstp]
- [edit routing-instances *routing-instance-name* protocols vstp]

2. Revert the forced IEEE 802.1D STP to run as the configured RSTP or VSTP:

```
user@host# clear spanning-tree protocol-migration <interface interface-name>  
<routing-instance routing-instance-name>
```

To revert the STP protocol for the specified interface only, specify the **interface *interface-name*** option.

To revert the STP protocol for a particular routing instance only, specify the **routing-instance *routing-instance-name*** option.

- Related Documentation**
- Spanning-Tree Protocols Supported on MX Series Routers on page 205
 - RSTP or VSTP Forced to Run as IEEE 802.1D STP on page 210
 - Configuring Rapid Spanning-Tree Protocol on page 217
 - Configuring VLAN Spanning-Tree Protocol on page 226

Basic Configuration of Bridges in Spanning-Tree Instances

The following topics describe the basic features you can configure for spanning-tree protocol instances:

- Provider Bridge Participation in RSTP or MSTP Instances on page 211
- System Identifier for Bridges in STP or RSTP Instances on page 212
- Bridge Priority for Election of Root Bridge and Designated Bridge on page 212
- Maximum Age for Awaiting Arrival of Hello BPDUs on page 212
- Hello Time for Root Bridge to Transmit Hello BPDUs on page 212
- Forward Delay Before Ports Transition to Forwarding State on page 213

Provider Bridge Participation in RSTP or MSTP Instances

A provider network can bridge the customer STP BPDU packets between customer sites by default. At the same time, the provider network can prevent forwarding loops by running a spanning-tree protocol in the provider network. On an MX Series router running Rapid Spanning-Tree Protocol (RSTP) or Multiple Spanning-Tree Protocol (MSTP) in a provider network, you can enable provider bridge participation in the RSTP or MSTP instance.

The IEEE 802.1ad specification reserves the group MAC address value of **01:80:c2:00:00:08** to designate the *provider bridge group*. On an MX Series router for which you have enabled provider bridge participation in the RSTP or MSTP instance, the router exchanges BPDU packets with the provider bridge group as follows:

- Transmitted BPDU packets contain the destination MAC address **01:80:c2:00:00:08**.
- Received BPDU packets with the destination MAC address **01:80:c2:00:00:08** are accepted and passed to the Routing Engine.

- Related Documentation**
- Spanning-Tree Protocols Supported on MX Series Routers on page 205
 - Configuring Rapid Spanning-Tree Protocol on page 217
 - Configuring Multiple Spanning-Tree Protocol on page 220
 - **bpdudestinationmacaddress** on page 262

System Identifier for Bridges in STP or RSTP Instances

The extended system identifier is used to specify different bridge identifiers for different STP or RSTP routing instances.

- Related Documentation**
- [Configuring Rapid Spanning-Tree Protocol on page 217](#)
 - [extended-system-id on page 269](#)

Bridge Priority for Election of Root Bridge and Designated Bridge

Use the bridge priority to control which bridge is elected as the root bridge and also to control which bridge is elected the root bridge when the initial root bridge fails.

The root bridge for each spanning-tree protocol instance is determined by the bridge ID. The bridge ID consists of a configurable bridge priority and the MAC address of the bridge. The bridge with the lowest bridge ID is elected as the root bridge. If the bridge priorities are equal or if the bridge priority is not configured, the bridge with the lowest MAC address is elected the root bridge.

The bridge priority can also be used to determine which bridge becomes the designated bridge for a LAN segment. If two bridges have the same path cost to the root bridge, the bridge with the lowest bridge ID becomes the designated bridge.

The bridge priority can be set only in increments of 4096.

- Related Documentation**
- [Configuring Rapid Spanning-Tree Protocol on page 217](#)
 - [Configuring Multiple Spanning-Tree Protocol on page 220](#)
 - [Configuring VLAN Spanning-Tree Protocol on page 226](#)
 - [bridge-priority on page 264](#)

Maximum Age for Awaiting Arrival of Hello BPDUs

The maximum age timer specifies the maximum expected arrival time of hello BPDUs. If the maximum age timer expires, the bridge detects that the link to the root bridge has failed and initiates a topology reconvergence. The maximum age timer should be longer than the configured hello timer.

- Related Documentation**
- [Configuring Rapid Spanning-Tree Protocol on page 217](#)
 - [Configuring Multiple Spanning-Tree Protocol on page 220](#)
 - [Configuring VLAN Spanning-Tree Protocol on page 226](#)
 - [max-age on page 275](#)

Hello Time for Root Bridge to Transmit Hello BPDUs

The hello timer specifies the time interval at which the root bridge transmits configuration BPDUs.

- Related Documentation**
- [Configuring Rapid Spanning-Tree Protocol on page 217](#)
 - [Configuring Multiple Spanning-Tree Protocol on page 220](#)
 - [Configuring VLAN Spanning-Tree Protocol on page 226](#)
 - [hello-time on page 271](#)

Forward Delay Before Ports Transition to Forwarding State

The forwarding delay timer specifies the length of time a spanning-tree protocol bridge port remains in the listening and learning states before transitioning to the forwarding state. Setting the interval too short could cause unnecessary spanning-tree reconvergence. Before changing this parameter, you should have a thorough understanding of spanning-tree protocols.

- Related Documentation**
- [Configuring Rapid Spanning-Tree Protocol on page 217](#)
 - [Configuring Multiple Spanning-Tree Protocol on page 220](#)
 - [Configuring VLAN Spanning-Tree Protocol on page 226](#)
 - [forward-delay on page 270](#)

Physical Interface Configuration for Spanning-Tree Protocol Instances

The following topics describe guidelines for configuring spanning-tree protocol instance interfaces:

- [Spanning-Tree Instance Interface on page 213](#)
- [Spanning-Tree Instance Interface Priority on page 214](#)
- [Spanning-Tree Instance Interface Cost on page 214](#)
- [Spanning-Tree Instance Interface Point-to-Point Link Mode on page 215](#)
- [Spanning-Tree Instance Interface Configured as an Edge Port on page 215](#)

Spanning-Tree Instance Interface

STP and RSTP are limited to a single instance on any physical interface. Use the **interface** statement to configure which interfaces participate in the STP or RSTP instance.

MSTP supports multiple instances on a single physical interface. Use the **interface** statement to configure which logical interfaces participate in MSTP.

For VSTP, interfaces can be configured at the global level or at the VLAN level. Interfaces configured at the global VSTP level will be enabled for all the configured VLANs. If an interface is configured at both the global and VLAN levels, the configuration at the VLAN level overrides the global configuration.

- Related Documentation**
- [Configuring Rapid Spanning-Tree Protocol on page 217](#)
 - [Configuring Multiple Spanning-Tree Protocol on page 220](#)
 - [Configuring VLAN Spanning-Tree Protocol on page 226](#)

- [cost on page 266](#)
- [edge on page 268](#)
- [interface \(Spanning Tree\) on page 273](#)
- [mode on page 277](#)
- [priority on page 281](#)

Spanning-Tree Instance Interface Priority

The root port is the interface on the nonroot bridge with the lowest path cost to the root bridge. When multiple interfaces have the same path cost to the root bridge, the interface with the lowest interface priority is selected as the root port.

If the interface priority is not configured and multiple interfaces have the same path cost to the root bridge, the interface with the lowest interface identifier is selected as the root port.

If the interface priority is configured under the MSTP protocol, this becomes the default value for all interfaces. If the interface priority is configured under the MSTI interface, the value overrides the default for that interface.

If the interface priority is configured at both the VSTP global and VLAN levels, the configuration at the VLAN level overrides the global configuration.

Related Documentation

- [Configuring Rapid Spanning-Tree Protocol on page 217](#)
- [Configuring Multiple Spanning-Tree Protocol on page 220](#)
- [Configuring VLAN Spanning-Tree Protocol on page 226](#)
- [interface \(Spanning Tree\) on page 273](#)
- [priority on page 281](#)

Spanning-Tree Instance Interface Cost

The path cost used to calculate the root path cost from any given LAN segment is determined by the total cost of each link in the path. By default, the link cost is determined by the speed of the link. The interface cost can be configured to override the default cost and control which bridge is the designated bridge and which port is the designated port. In MSTP the CIST external path cost is determined by the link speed and the number of hops.

If the interface cost is not configured, the cost is determined by the speed of the interface. For example, a 100-Mbps link has a default path cost of 19, a 1000-Mbps link has a default path cost of 4, and a 10-Gbps link has a default path cost of 2.

If the interface cost is configured under MSTP, this becomes the default value for all interfaces. If the interface cost is configured under the MSTI interface, the value overrides the default for that interface.

If the interface cost is configured at both the VSTP global and VLAN levels, the configuration at the VLAN level overrides the global configuration.

The interface cost should be set the same for all interfaces connected to the same LAN segment.

Related Documentation

- Configuring Rapid Spanning-Tree Protocol on page 217
- Configuring Multiple Spanning-Tree Protocol on page 220
- Configuring VLAN Spanning-Tree Protocol on page 226
- **cost on page 266**
- **interface (Spanning Tree) on page 273**

Spanning-Tree Instance Interface Point-to-Point Link Mode

The interface mode allows RSTP, MSTP, and VSTP to converge faster than the original STP on point-to-point links. The protocol does not need to wait for timers on point-to-point links. Configure interfaces that have a point-to-point link to another Layer 2 bridge as **p2p**. This parameter is ignored if the STP is configured to run the original spanning-tree version.

If the interface mode is configured at both the VSTP global and VLAN levels, the configuration at the VLAN level overrides the global configuration.

Related Documentation

- Configuring Rapid Spanning-Tree Protocol on page 217
- Configuring Multiple Spanning-Tree Protocol on page 220
- Configuring VLAN Spanning-Tree Protocol on page 226
- **mode on page 277**
- **interface (Spanning Tree) on page 273**

Spanning-Tree Instance Interface Configured as an Edge Port

RSTP, MSTP, and VSTP instance interfaces configured as *edge ports* enable the protocol to converge faster than the original IEEE 802.1D STP version. Edge ports transition directly to the forwarding state, and so the protocol does not need to wait for BPDUs to be received on edge ports.

The Junos OS supports automatic detection of edge ports as described in the RSTP standard. Layer 2 bridges do not expect to receive BPDUs for edge ports. If a BPDU is received for an edge port, the port becomes a non-edge port.

Keep the following guidelines in mind when configuring spanning-tree instance interfaces as edge ports:

- Do not configure a spanning-tree instance interface as an edge port if it is connected to any Layer 2 bridge. An instance interface connected to Layer 2 bridges but configured as an edge port can cause physical loops.

- if the spanning-tree protocol is configured to run the original IEEE 802.1D spanning-tree version, the edge-port option (if configured) is ignored.
- If edge ports are configured at both the VSTP global and VLAN levels, the configuration at the VLAN level overrides the global configuration.

**Related
Documentation**

- [Configuring Rapid Spanning-Tree Protocol on page 217](#)
- [Configuring Multiple Spanning-Tree Protocol on page 220](#)
- [Configuring VLAN Spanning-Tree Protocol on page 226](#)
- [edge on page 268](#)
- [interface \(Spanning Tree\) on page 273](#)

Spanning-Tree Protocol Trace Options

In order to trace spanning-tree protocol operations, you can set spanning-tree protocol-specific trace options in the spanning-tree protocol configuration.

For general information about tracing and global tracing options, see the statement summary for the global **traceoptions** statement in the [Junos OS Routing Protocols Configuration Guide](#).

**Related
Documentation**

- [Configuring Rapid Spanning-Tree Protocol on page 217](#)
- [Configuring Multiple Spanning-Tree Protocol on page 220](#)
- [Configuring VLAN Spanning-Tree Protocol on page 226](#)
- [Example: Tracing Spanning-Tree Protocol Operations on page 256](#)
- [traceoptions \(Spanning Tree\) on page 288](#)

Configuring Spanning-Tree Protocols

- Configuring Rapid Spanning-Tree Protocol on page 217
- Configuring Multiple Spanning-Tree Protocol on page 220
- Configuring VLAN Spanning-Tree Protocol on page 226
- Tracing Spanning-Tree Operations on page 230

Configuring Rapid Spanning-Tree Protocol

You can configure Rapid Spanning-Tree Protocol (RSTP) under the following hierarchy levels:

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

The routing instance type can be either **virtual-switch** or **layer2-control**.

To configure the Rapid Spanning-Tree Protocol:

1. Enable RSTP as the version of spanning-tree protocol to be configured:

```
[edit]
user@host@ edit ... protocols rstp
```

2. (Optional) For compatibility with older bridges that do not support RSTP, you can run force RSTP to run as the original IEEE 802.1D Spanning-Tree Protocol (STP) version:

```
[edit ... protocols rstp]
user@host# set force-version stp
```



NOTE: If RSTP has been forced to run as the original STP version, you can revert back to RSTP by first removing the **force-version** statement from the configuration and then entering the **clear spanning-tree protocol-migration** configuration mode command.

3. (Optional) Enable provider bridge participation in the RSTP instance:

```
[edit ... protocols rstp]
user@host# set bpdu-destination-mac-address provider-bridge-group
```

4. (Optional) Specify the extended system identifier used in identifiers bridges that participate in RSTP:

```
[edit ... protocols rstp]
user@host# set extended-system-id identifier
```

5. Configure the interfaces that participate in the RSTP instance.

- a. Enable configuration of the interface:

```
[edit ... protocols rstp]
user@host# edit interface interface-name
```

- b. Configure the interface priority:

```
[edit ... protocols rstp interface interface-name]
user@host# set priority interface-priority
```

- c. (Optional) By default, the interface link cost is determined by the link speed. You can configure the interface link cost to control which bridge is the designated bridge and which port is the designated port:

```
[edit ... protocols rstp interface interface-name]
user@host# set cost interface-link-cost
```

- d. Configure the interface link mode to identify point-to-point links:

```
[edit ... protocols rstp interface interface-name]
user@host# set mode (p2p | shared)
```

Specify **p2p** if the link is point to point. Specify **shared** if the link is a shared media.

- e. (Optional) Configure the interface as an edge port:

```
[edit ... protocols rstp interface interface-name]
user@host# set edge
```

Edge ports do not expect to receive bridge protocol data unit (BPDU) packets. If a BPDU packet is received for an edge port, the port becomes a nonedge port.

You can also enable BPDU root protection for all spanning-tree protocol instances on the interface. BPDU root protect ensures the port is the spanning-tree designated port. If the port receives superior BPDU packets, root protect moves this port to a root-prevented spanning-tree state. For configuration details, see “Checking the Status of Spanning-Tree Instance Interfaces” on page 241.

6. Configure the bridge priority

```
[edit ... protocols rstp]
user@host# set bridge-priority bridge-priority
```

For more information, see “Bridge Priority for Election of Root Bridge and Designated Bridge” on page 212.

7. Configure hello BPDU timers.

- a. Configure the maximum expected arrival time of hello BPDUs:

```
[edit ... protocols rstp]
user@host# set max-age seconds
```

- b. Configure the time interval at which the root bridge transmits configuration BPDUs:

```
[edit ... protocols rstp]
user@host# set hello-time seconds
```

8. (Optional) By default, the bridge port remains in the listening and learning states for 15 seconds before transitioning to the forwarding state. You can specify a delay from 4 through 20 seconds instead:

```
[edit ... protocols rstp]
user@host# set forward-delay seconds
```

9. Verify the RSTP configuration:

```
[edit]
... { # Optional logical system and/or routing instance
  protocols {
    rstp {
      force-version stp; # Optional.
      bpdu-destination-mac-address provider-bridge-group; # Optional
      extended-system-id identifier; # Optional.
      interface interface-name {
        priority interface-priority;
        cost interface-link-cost; # Optional.
        mode (p2p | shared);
        edge; # Optional.
      }
      bridge-priority bridge-priority;
      max-age seconds;
      hello-time seconds;
      forward-delay seconds; # Optional.
    }
  }
}
```

Related Documentation

- Spanning-Tree Protocols Supported on MX Series Routers on page 205
- RSTP or VSTP Forced to Run as IEEE 802.1D STP on page 210
- Reverting RSTP or VSTP Back From Forced IEEE 802.1D STP on page 210
- Provider Bridge Participation in RSTP or MSTP Instances on page 211
- System Identifier for Bridges in STP or RSTP Instances on page 212

Configuring Multiple Spanning-Tree Protocol

The following topics describe Multiple Spanning-Tree Protocol configuration and operation tasks on MX Series routers:

- Configuring Multiple Spanning-Tree Protocol on page 220
- Configuring MST Instances on a Physical Interface on page 224
- Disabling MSTP on page 225

Configuring Multiple Spanning-Tree Protocol

You can configure the Multiple Spanning-Tree Protocol (MSTP) under the following hierarchy levels:

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

The routing instance type can be either **virtual-switch** or **layer2-control**.

To configure the Multiple Spanning-Tree Protocol:

1. Enable RSTP as the version of spanning-tree protocol to be configured:

```
[edit]
user@host@ edit ... protocols mstp
```

2. (Optional) Enable provider bridge participation in the MSTP instance:

```
[edit ... protocols mstp]
user@host# set bpdu-destination-mac-address provider-bridge-group
```


3. Configure the interfaces that participate in the MSTP instance.

a. Enable configuration of the interface:

```
[edit ... protocols mstp]
user@host# edit interface interface-name
```

b. Configure the interface priority:

```
[edit ... protocols mstp interface interface-name]
user@host# set priority interface-priority
```

c. (Optional) By default, the interface link cost is determined by the link speed. You can configure the interface link cost to control which bridge is the designated bridge and which port is the designated port:

```
[edit ... protocols mstp interface interface-name]
user@host# set cost interface-link-cost
```

d. Configure the interface link mode to identify point-to-point links:

```
[edit ... protocols mstp interface interface-name]
user@host# set mode (p2p | shared)
```

Specify **p2p** if the link is point to point. Specify **shared** if the link is a shared media.

e. (Optional) Configure the interface as an edge port:

```
[edit ... protocols mstp interface interface-name]
user@host# set edge
```

Edge ports do not expect to receive bridge protocol data unit (BPDU) packets. If a BPDU packet is received for an edge port, the port becomes a nonedge port.

You can also enable BPDU root protection for all spanning-tree protocol instances on the interface. BPDU root protect ensures the port is the spanning-tree designated port. If the port receives superior BPDU packets, root protect moves this port to a root-prevented spanning-tree state. For configuration details, see “Checking the Status of Spanning-Tree Instance Interfaces” on page 241.

4. Configure the bridge priority

```
[edit ... protocols mstp]
user@host# set bridge-priority bridge-priority
```

For more information, see “Bridge Priority for Election of Root Bridge and Designated Bridge” on page 212.

5. Configure hello BPDU timers.

- a. Configure the maximum expected arrival time of hello BPDUs:

```
[edit ... protocols mstp]
user@host# set max-age seconds
```

- b. Configure the time interval at which the root bridge transmits configuration BPDUs:

```
[edit ... protocols mstp]
user@host# set hello-time seconds
```

6. (Optional) By default, the bridge port remains in the listening and learning states for 15 seconds before transitioning to the forwarding state. You can specify a delay from 4 through 20 seconds instead:

```
[edit ... protocols mstp]
user@host# set forward-delay seconds
```

7. Configure MSTP-specific options.

- a. Configure the MSTP region configuration name:

```
[edit ... protocols mstp]
user@host# set configuration-name configuration-name
```

- b. Configure the MSTP revision level:

```
[edit ... protocols mstp]
user@host# set revision-level revision-level
```

- c. Configure the maximum number of hops a BPDU can be forwarded in the MSTP region:

```
[edit ... protocols mstp]
user@host# set max-hops hops
```

8. Verify the MSTP configuration:

```
[edit]
... { # Optional logical system and/or routing instance
  protocols {
    mstp {
      bpd-destination-mac-address provider-bridge-group; # Optional
      interface interface-name {
        priority interface-priority;
        cost interface-link-cost; # Optional.
        mode (p2p | shared);
        edge; # Optional.
      }
      bridge-priority bridge-priority;
      max-age seconds;
      hello-time seconds;
      forward-delay seconds; # Optional.
      configuration-name configuration-name; # MST region configuration name.
      revision-level revision-level; # MST revision number.
      max-hops hops; # MST maximum hops.
    }
  }
}
```

}

**Related
Documentation**

- [Spanning-Tree Protocols Supported on MX Series Routers on page 205](#)
- [Configuring MST Instances on a Physical Interface on page 224](#)
- [Disabling MSTP on page 225](#)

Configuring MST Instances on a Physical Interface

You can configure a Multiple Spanning-Tree Instance (MSTI) under the following hierarchy levels:

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

The routing instance type can be either **virtual-switch** or **layer2-control**.

Before you begin, configure Multiple Spanning-Tree Protocol. For configuration details, see “Configuring MSTP” on page 220.

1. Enable configuration of a MST instance:

```
[edit]
user@host# edit ... protocols mstp msti msti-id
```

The *msti-id* value must be from 1 through 64.

2. Configure the interfaces that participate in the MST instance.

- a. Enable configuration of the interface:

```
[edit ... protocols mstp msti msti-id]
user@host# edit interface interface-name
```

- b. Configure the interface priority:

```
[edit ... protocols mstp msti msti-id interface interface-name]
user@host# set priority interface-priority
```

- c. (Optional) By default, the interface link cost is determined by the link speed. You can configure the interface link cost to control which bridge is the designated bridge and which port is the designated port:

```
[edit ... protocols mstp msti msti-id interface interface-name]
user@host# set cost interface-link-cost
```

- d. (Optional) Configure the interface as an edge port:

```
[edit ... protocols mstp msti msti-id interface interface-name]
user@host# set edge
```

Edge ports do not expect to receive bridge protocol data unit (BPDU) packets. If a BPDU packet is received for an edge port, the port becomes a nonedge port

3. Configure the bridge priority

```
[edit ... protocols mstp msti msti-id]
user@host# set bridge-priority bridge-priority
```

For more information, see “Bridge Priority for Election of Root Bridge and Designated Bridge” on page 212.

4. (Optional) An MSTI can map to a range of VLANs just as a logical port can map to a range of VLANs. The MSTP VLAN specifies the VLAN or VLAN range to which this MSTI is mapped. The `vlan-id` is configured under the logical interface. Configure the VLAN or VLAN range of the MSTI instance:

```
[edit]
user@host# set vlan (vlan-id | vlan-id-range)
```

5. Verify the MST interface configuration.

```
[edit]
protocols {
  mstp {
    ...basic-mstp-configuration...
    msti msti-id { # Instance identifier 1 – 64.
      bridge-priority priority;
      vlan vlan-id; # Optional
      interface interface-name {
        cost cost;
        edge;
        priority interface-priority;
      }
    }
  }
}
```

Related Documentation

- Spanning-Tree Protocols Supported on MX Series Routers on page 205
- Configuring Multiple Spanning-Tree Protocol on page 220
- Disabling MSTP on page 225

Disabling MSTP

To disable the entire MSTP instance:

- Include the **disable** statement. You can include this statement at the following hierarchy levels:
 - [edit logical-systems *logical-system-name* protocols mstp]
 - [edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols mstp]
 - [edit protocols mstp]
 - [edit routing-instances *routing-instance-name* protocols mstp]

Related Documentation

- Spanning-Tree Protocols Supported on MX Series Routers on page 205
- Configuring Multiple Spanning-Tree Protocol on page 220
- Configuring MST Instances on a Physical Interface on page 224

Configuring VLAN Spanning-Tree Protocol

You can configure the VLAN Spanning-Tree Protocol (VSTP) under the following hierarchy levels:

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

The routing instance type can be either **virtual-switch** or **layer2-control**.

To configure the VLAN Spanning-Tree Protocol:

1. Enable VSTP as the version of spanning-tree protocol to be configured:

```
[edit]
user@host@ edit ... protocols vstp
```

2. (Optional) For compatibility with older bridges that do not support VSTP, you can run force VSTP to run as the original IEEE 802.1D Spanning-Tree Protocol (STP) version:

```
[edit ... protocols vstp]
user@host# set force-version stp
```



NOTE: If VSTP has been forced to run as the original STP version, you can revert back to VSTP by first removing the **force-version** statement from the configuration and then entering the **clear spanning-tree protocol-migration** configuration mode command.

3. Configure the interfaces that participate in the VSTP instance.

a. Enable configuration of the interface:

```
[edit ... protocols vstp]
user@host# edit interface interface-name
```

b. Configure the interface priority:

```
[edit ... protocols vstp interface interface-name]
user@host# set priority interface-priority
```

c. (Optional) By default, the interface link cost is determined by the link speed. You can configure the interface link cost to control which bridge is the designated bridge and which port is the designated port:

```
[edit ... protocols vstp interface interface-name]
user@host# set cost interface-link-cost
```

d. Configure the interface link mode to identify point-to-point links:

```
[edit ... protocols vstp interface interface-name]
user@host# set mode (p2p | shared)
```

Specify **p2p** if the link is point to point. Specify **shared** if the link is a shared media.

e. (Optional) Configure the interface as an edge port:

```
[edit ... protocols vstp interface interface-name]
user@host# set edge
```

Edge ports do not expect to receive bridge protocol data unit (BPDU) packets. If a BPDU packet is received for an edge port, the port becomes a nonedge port.

You can also enable BPDU root protection for all spanning-tree protocol instances on the interface. BPDU root protect ensures the port is the spanning-tree designated port. If the port receives superior BPDU packets, root protect moves this port to a root-prevented spanning-tree state. For configuration details, see “Checking the Status of Spanning-Tree Instance Interfaces” on page 241.

4. Enable configuration of a VLAN instance:

```
[edit ... protocols vstp]
user@host# edit vlan vlan-id
```

5. Configure the bridge priority

```
[edit ... protocols vstp vlan vlan-id]
user@host# set bridge-priority bridge-priority
```

For more information, see “Bridge Priority for Election of Root Bridge and Designated Bridge” on page 212.

6. Configure hello BPDU timers.

- a. Configure the maximum expected arrival time of hello BPDUs:

```
[edit ... protocols vstp vlan vlan-id]  
user@host# set max-age seconds
```

- b. Configure the time interval at which the root bridge transmits configuration BPDUs:

```
[edit ... protocols vstp vlan vlan-id]  
user@host# set hello-time seconds
```

7. (Optional) By default, the bridge port remains in the listening and learning states for 15 seconds before transitioning to the forwarding state. You can specify a delay from 4 through 20 seconds instead:

```
[edit ... protocols vstp vlan vlan-id]  
user@host# set forward-delay seconds
```


8. Configure the interfaces that participate in the VSTP instance.

a. Enable configuration of the interface:

```
[edit ... protocols vstp vlan vlan-id]
user@host# edit interface interface-name
```

b. Configure the interface priority:

```
[edit ... protocols vstp vlan vlan-id interface interface-name]
user@host# set priority interface-priority
```

c. (Optional) By default, the interface link cost is determined by the link speed. You can configure the interface link cost to control which bridge is the designated bridge and which port is the designated port:

```
[edit ... protocols vstp vlan vlan-id interface interface-name]
user@host# set cost interface-link-cost
```

d. Configure the interface link mode to identify point-to-point links:

```
[edit ... protocols vstp vlan vlan-id interface interface-name]
user@host# set mode (p2p | shared)
```

Specify **p2p** if the link is point to point. Specify **shared** if the link is a shared media.

e. (Optional) Configure the interface as an edge port:

```
[edit ... protocols vstp vlan vlan-id interface interface-name]
user@host# set edge
```

Edge ports do not expect to receive bridge protocol data unit (BPDU) packets. If a BPDU packet is received for an edge port, the port becomes a nonedge port.

You can also enable BPDU root protection for all spanning-tree protocol instances on the interface. BPDU root protect ensures the port is the spanning-tree designated port. If the port receives superior BPDU packets, root protect moves this port to a root-prevented spanning-tree state. For configuration details, see “Checking the Status of Spanning-Tree Instance Interfaces” on page 241.

9. Verify the VSTP configuration:

```
[edit]
... { # Optional logical system and/or routing instance
  protocols {
    vstp {
      force-version stp; # Optional.
      interface interface-name {
        priority interface-priority;
        cost interface-link-cost; # Optional.
        mode (p2p | shared);
        edge; # Optional.
      }
    }
    vlan vlan-id {
      bridge-priority bridge-priority;
      max-age seconds;
      hello-time seconds;
      forward-delay seconds; # Optional.
    }
  }
}
```

```

        interface interface-name {
            priority interface-priority;
            cost interface-link-cost; # Optional.
            mode (p2p | shared);
            edge; # Optional.
        }
    }
}

```

Related Documentation

- Spanning-Tree Protocols Supported on MX Series Routers on page 205
- RSTP or VSTP Forced to Run as IEEE 802.1D STP on page 210
- Reverting RSTP or VSTP Back From Forced IEEE 802.1D STP on page 210
- VPLS Multihomed Layer 2 Ring and MPLS Infrastructure Overview on page 242
- VPLS Multihomed Layer 2 Ring and MPLS Infrastructure Topology on page 243

Tracing Spanning-Tree Operations

You can enable global routing protocol tracing options at the **[edit routing-options] Hierarchy Level**. For general information about tracing and global tracing options, see the statement summary for the global traceoptions statement in the [Junos OS Routing Protocols Configuration Guide](#).

In addition, you can enable STP-specific trace options at the following hierarchy levels:

- **[edit logical-systems logical-system-name protocols (mstp | rstp | vstp)]**
- **[edit logical-systems logical-system-name routing-instances routing-instance-name protocols (mstp | rstp | vstp)]**
- **[edit protocols (mstp | rstp | vstp)]**
- **[edit routing-instances routing-instance-name protocols (mstp | rstp | vstp)]**

The routing instance type can be either **virtual-switch** or **layer2-control**.

To enable tracing of spanning-tree protocol operations:

1. Enable configuration of the spanning-tree protocol whose operations are to be traced:

```

[edit]
user@host# edit ... protocols (mstp | rstp | vstp)

```

2. Enable configuration of spanning-tree protocol-specific trace options:

```

[edit ... protocols (mstp | rstp | vstp)]
user@host# edit traceoptions

```

3. Configure the files that contain trace logging information:

```

[edit ... protocols (mstp | rstp | vstp)]

```

```
user@host# set file filename <files number> <size bytes>
<world-readable | no-world-readable>
```

4. Configure spanning-tree protocol-specific options.

- a. To enable a spanning-tree protocol-specific option, include the **flag** statement:

```
[edit ... protocols (mstp | rstp | vstp)]
user@host# set flag flag <flag-modifier> <disable>
```

You can specify the following spanning-tree protocol-specific **flag** options:

- **all**—Trace all operations.
- **all-failures**—Trace all failure conditions.
- **bpdu**—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-transmit-state-machine**—Trace the port transmit state machine.
- **port-state-transit-state-machine**—Trace the port state transit 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.



NOTE: Use the trace flag **all** with caution. This flag may cause the CPU to become very busy.

- b. To disable an individual spanning-tree protocol-specific option, include the **disable** option with the **flag** statement.

5. Verify the spanning-tree protocol-specific trace options.

```
[edit]
...
routing-options {
```

```
    traceoptions {  
        ...global-trace-options-configuration...  
    }  
}  
protocols {  
    (mstp | rstp | vstp) {  
        traceoptions { # Spanning-tree protocol-specific.  
            file filename <files number> <size bytes> <world-readable | no-world-readable>;  
            flag flag <flag-modifier> <disable>;  
        }  
    }  
}  
...
```

**Related
Documentation**

- [Spanning-Tree Protocols Supported on MX Series Routers on page 205](#)
- [Example: Tracing Spanning-Tree Protocol Operations on page 256](#)

Spanning-Tree Protocol Options

- [Loop Protection for Spanning-Tree Instance Interfaces on page 233](#)
- [Root Protect for Spanning-Tree Instance Interfaces on page 236](#)
- [BPDU Protection for Spanning-Tree Instance Interfaces on page 237](#)
- [VPLS Root Protection Topology Change Actions on page 242](#)
- [Layer 2 Protocol Tunneling Through a Network on page 250](#)

Loop Protection for Spanning-Tree Instance Interfaces

The following topics describe loop protection for spanning-tree instance interfaces:

- [Loop Protection for Spanning-Tree Instance Interfaces Overview on page 233](#)
- [Loop Protection for a Spanning-Tree Instance Interface on page 234](#)
- [Configuring Loop Protection for a Spanning-Tree Instance Interface on page 235](#)

Loop Protection for Spanning-Tree Instance Interfaces Overview

Spanning-tree protocol loop protection enhances the normal checks that spanning-tree protocols perform on interfaces. Loop protection performs a specified action when BPDUs are not received on a nondesignated port interface. You can choose to block the interface or issue an alarm when bridge protocol data units (BPDUs) are not received on the port.

The spanning-tree protocol family is responsible for breaking loops in a network of bridges with redundant links. However, hardware failures can create forwarding loops (STP loops) and cause major network outages. Spanning-tree protocols break loops by blocking ports (interfaces). However, errors occur when a blocked port transitions erroneously to a forwarding state.

Ideally, a spanning-tree protocol bridge port remains blocked as long as a superior alternate path to the root bridge exists for a connected LAN segment. This designated port is determined by receiving superior BPDUs from a peer on that port. When other ports no longer receive BPDUs, the spanning-tree protocol considers the topology to be loop free. However, if a blocked or alternate port moves into a forwarding state, this creates a loop.

By default (that is, without spanning-tree protocol loop protection configured), an interface that stops receiving BPDUs will assume the designated port role and possibly result in a spanning-tree protocol loop.

You can configure spanning-tree protocol loop protection to improve the stability of Layer 2 networks.

You configure spanning-tree protocol loop protection to prevent selected interfaces from interpreting the lack of received BPDUs as a “false positive” condition for making the interface the designated port.

**Related
Documentation**

- Loop Protection for a Spanning-Tree Instance Interface on page 234
- Configuring Loop Protection for a Spanning-Tree Instance Interface on page 235
- Example: Enabling Loop Protection for Spanning-Tree Protocols on page 255

Loop Protection for a Spanning-Tree Instance Interface

By default, a spanning-tree protocol interface that stops receiving Bridge Protocol Data Unit (BPDU) data frames will transition to the designated port (forwarding) state, creating a potential loop. To prevent a spanning-tree instance interface from interpreting a lack of received BPDUs as a “false positive” condition for assuming the designated port role, you can configure one of the following loop protection options:

- Configure the router to raise an alarm condition if the spanning-tree instance interface has not received BPDUs during the timeout interval.
- Configure the router to block the spanning-tree instance interface if the interface has not received BPDUs during the timeout interval.



NOTE: Spanning-tree instance interface loop protection is enabled for all spanning-tree instances on the interface, but blocks or alarms only those instances that stop receiving BPDUs.

We recommend you configure loop protection only on non-designated interfaces such as the root or alternate interfaces. Otherwise, if you configure loop protection on both sides of a designated link, then certain STP configuration events (such as setting the root bridge priority to an inferior value in a topology with many loops) can cause both interfaces to transition to blocking mode.

**Related
Documentation**

- Loop Protection for Spanning-Tree Instance Interfaces Overview on page 233
- Configuring Loop Protection for a Spanning-Tree Instance Interface on page 235
- Example: Enabling Loop Protection for Spanning-Tree Protocols on page 255
- **bpdu-timeout-action** on page 263
- **interface (Spanning Tree)** on page 273

Configuring Loop Protection for a Spanning-Tree Instance Interface

Before you begin, you must fully configure the spanning-tree protocol, including instance interfaces. You can configure RSTP, MSTP, or VSTP at the following hierarchy levels:

- [edit protocols]
- [edit routing-instances *routing-instance-name* protocols]

To configure enhanced loop protection:

1. Include the **bpdu-timeout-action** statement with either the **block** or **log** option for the spanning-tree protocol interface.
 - For the STP or RSTP instance on a physical interface:

```
[edit]
protocols {
  rstp {
    interface interface-name {
      bpdu-timeout-action (log | block);
    }
  }
}
```

- For all MSTP instances on a physical interface:

```
[edit]
protocols {
  mstp {
    interface interface-name {
      bpdu-timeout-action (log | block);
    }
  }
}
```

- For all VSTP instances on a physical interface configured at the global level or a the VLAN level:

```
[edit]
protocols {
  vstp {
    interface interface-name {
      bpdu-timeout-action (log | block);
    }
    vlan vlan-id {
      interface interface-name {
        bpdu-timeout-action (log | block);
      }
    }
  }
}
```

2. To display the spanning-tree protocol loop protection characteristics on an interface, use the `show spanning-tree interface operational` command.

- Related Documentation**
- [Loop Protection for Spanning-Tree Instance Interfaces Overview on page 233](#)
 - [Loop Protection for a Spanning-Tree Instance Interface on page 234](#)
 - [Example: Enabling Loop Protection for Spanning-Tree Protocols on page 255](#)

Root Protect for Spanning-Tree Instance Interfaces

The following topics describe root protect for spanning-tree instance interfaces:

- [Root Protection for Spanning-Tree Instance Interfaces Overview on page 236](#)
- [Root Protect for a Spanning-Tree Instance Interface on page 236](#)
- [Enabling Root Protect for a Spanning-Tree Instance Interface on page 237](#)

Root Protection for Spanning-Tree Instance Interfaces Overview

Root protect helps to enforce the root bridge placement in a Layer 2 switched network. Enable root protect on interfaces that should not receive superior bridge protocol data units (BPDUs) from the root bridge. Typically, these ports are Spanning-Tree-Protocol-designated ports on an administrative boundary. Enabling root protect ensures the port remains a spanning-tree designated port.

If the bridge receives superior BPDUs on a port that has root protect enabled, that port transitions to a root-prevented STP state and the interface is blocked. This prevents a bridge that should not be the root bridge from being elected the root bridge.

After the bridge stops receiving superior BPDUs on the port with root protect enabled and the received BPDUs time out, that port transitions back to the STP-designated port state.

- Related Documentation**
- [Root Protect for a Spanning-Tree Instance Interface on page 236](#)
 - [Enabling Root Protect for a Spanning-Tree Instance Interface on page 237](#)

Root Protect for a Spanning-Tree Instance Interface

When root protect is enabled on an interface, it is enabled for all spanning-tree protocol instances on that interface. The interface is blocked only for those instances that receive superior BPDUs.

By default, root protect is disabled.

- Related Documentation**
- [Root Protection for Spanning-Tree Instance Interfaces Overview on page 236](#)
 - [Enabling Root Protect for a Spanning-Tree Instance Interface on page 237](#)
 - [interface \(Spanning Tree\) on page 273](#)
 - [no-root-port on page 280](#)

Enabling Root Protect for a Spanning-Tree Instance Interface

To enable root protect for a spanning-tree instance interface:

1. Enable configuration of the spanning-tree protocol:

```
[edit]
user@host# edit protocols (mstp | rstp | vstp <vlan vlan-id>)
```

2. Enable configuration of the spanning-tree instance interface:

```
[edit ... protocols (mstp | rstp | vstp <vlan vlan-id>)]
user@host# edit interface interface-name
```

3. Enable root protection on the interface:

```
[edit ... protocols (mstp | rstp | vstp <vlan vlan-id>) interface interface-name]
user@host# set no-root-port
```

4. Verify the configuration of root protect for the spanning-tree instance interface:

```
[edit ... protocols (mstp | rstp | vstp <vlan vlan-id>) interface interface-name]
user@host# top
user@host# show ... protocols
```

```
...
(mstp | rstp | vstp <vlan vlan-id>) {
  interface interface-name {
    no-root-port;
  }
}
```



NOTE: This is not a complete configuration.

Related Documentation

- Root Protection for Spanning-Tree Instance Interfaces Overview on page 236
- Root Protect for a Spanning-Tree Instance Interface on page 236

BPDU Protection for Spanning-Tree Instance Interfaces

The following topics describe BPDU protection for spanning-tree instance interfaces:

- BPDU Protection for Spanning-Tree Instance Interfaces Overview on page 238
- BPDU Protection for Individual Spanning-Tree Instance Interfaces on page 238
- Configuring BPDU Protection on Individual Interfaces on page 239
- BPDU Protection on All Edge Ports of the Bridge on page 240
- Configuring BPDU Protection on All Edge Ports on page 240
- Checking the Status of Spanning-Tree Instance Interfaces on page 241
- Clearing the Blocked Status of a Spanning-Tree Instance Interface on page 241

BPDU Protection for Spanning-Tree Instance Interfaces Overview

By default, if a Bridge Protocol Data Unit (BPDU) data frame is received on a blocked interface, the system will disable the interface and stop forwarding frames out the interface until the interface is explicitly cleared.

The Spanning-Tree Protocol (STP) family is designed to break possible loops in a Layer 2 bridged network. Loop prevention avoids damaging broadcast storms that can potentially render the network useless. STP processes on bridges exchange BPDUs to determine the LAN topology, decide the root bridge, stop forwarding on some ports, and so on. However, a misbehaving user application or device can interfere with the operation of the STP protocols and cause network problems.

On the MX Series routers only, you can configure BPDU protection to ignore BPDUs received on interfaces where none should be expected (for example, a LAN interface on a network edge with no other bridges present). If a BPDU is received on a blocked interface, the interface is disabled and stops forwarding frames. By default, all BPDUs are accepted and processed on all interfaces.

You can configure BPDU protection on interfaces with the following encapsulation types:

- **ethernet-bridge**
- **ethernet-vpls**
- **extended-vlan-bridge**
- **vlan-vpls**
- **extended-vlan-vpls**

You can configure BPDU protection on individual interfaces or on all the edge ports of the bridge.

Related Documentation

- Spanning-Tree Protocols Supported on MX Series Routers on page 205
- Root Protection for Spanning-Tree Instance Interfaces Overview on page 236
- VPLS Multihomed Layer 2 Ring and MPLS Infrastructure Overview on page 242

BPDU Protection for Individual Spanning-Tree Instance Interfaces

To configure BPDU protection on one or more spanning-tree instance interfaces, include the **bpdu-block** statement:

```
bpdu-block {  
  interface interface-name;  
  disable-timeout seconds;  
}
```



NOTE: If you also include the optional **disable-timeout *seconds*** statement, *blocked interfaces* are automatically cleared after the specified time interval unless the interval is 0.

- Related Documentation**
- Root Protection for Spanning-Tree Instance Interfaces Overview on page 236
 - BPDU Protection for Spanning-Tree Instance Interfaces Overview on page 238
 - Configuring BPDU Protection on Individual Interfaces on page 239

Configuring BPDU Protection on Individual Interfaces

On MX Series routers, you can configure BPDU protection to ignore BPDU received on interfaces where none should be expected. If a BPDU is received on a blocked interface, the interface is disabled and stops forwarding frames. By default, all BPDUs are accepted and processed on all interfaces.

To configure BPDU protection for individual spanning-tree instance interfaces:

1. Enable BPDU protection on a specific spanning-tree instance interface:

```
[edit]
user@host# edit protocols layer2-control bpdv-block
user@host# set interface interface (aex | (ge-fpc/pic/port | xe-fpc/pic/port))
```

If a BPDU is received on the interface, the system will disable the interface and stop forwarding frames out the interface until the bridging process is restarted.

2. (Optional) Configure the amount of time the system waits before *automatically* unblocking this interface after it has received a BPDU.

```
[edit protocols layer2-control bpdv-block interface interface-name]
user@host# set disable-timeout seconds
```

The range of the *seconds* option value is from 10 through 3600 seconds (one hour). A *seconds* option value of 0 is allowed, but this results in the default behavior (the interface is blocked until the interface is cleared).

3. Verify the configuration of BPDU blocking for individual interfaces:

```
[edit]
interfaces {
  ge-fpc/pic/port { # VLAN encapsulation on Gigabit Ethernet.
    encapsulation (ethernet-bridge | extended-vlan-bridge | extended-vlan-vpls |
      vlan-vpls);
  }
  xe-fpc/pic/port { # VLAN encapsulation on 10-Gigabit Ethernet.
    encapsulation (ethernet-bridge | extended-vlan-bridge | extended-vlan-vpls |
      vlan-vpls);
  }
  ae-X { # VLAN encapsulation
    encapsulation (ethernet-vpls vlan-vpls); # on Aggregated Ethernet.
    ...
  }
  ae-X { # Extended VLAN encapsulation
    vlan-tagging; # on Aggregated Ethernet.
    encapsulation extended-vlan-vpls;
    unit logical-unit-number {
      vlan-id number;
      .....
    }
  }
}
```

```

    }
    .....
  }
}
protocols
  layer2-control {
    bpdv-block
      interface interface-name;
      disable-timeout seconds;
    }
  }
}

```

- Related Documentation**
- Root Protection for Spanning-Tree Instance Interfaces Overview on page 236
 - BPDU Protection for Spanning-Tree Instance Interfaces Overview on page 238
 - BPDU Protection for Individual Spanning-Tree Instance Interfaces on page 238

BPDU Protection on All Edge Ports of the Bridge

To configure edge port blocking for a particular STP family member, include the **bpdv-block-on-edge** statement for **mstp**, **rstp**, or **vstp**:

```

bpdv-block-on-edge;
interface interface-name;

```



NOTE: In contrast to BPDU protection configured on individual spanning-tree instance interfaces, BPDU protection configured on all edge ports of an entire spanning-tree protocol *disables designated edge ports* and does not enable them again.

- Related Documentation**
- Root Protection for Spanning-Tree Instance Interfaces Overview on page 236
 - BPDU Protection for Spanning-Tree Instance Interfaces Overview on page 238
 - Configuring BPDU Protection on All Edge Ports on page 240

Configuring BPDU Protection on All Edge Ports

On MX Series routers, you can configure BPDU protection to ignore BPDU received on interfaces where none should be expected. If a BPDU is received on a blocked interface, the interface is disabled and stops forwarding frames. By default, all BPDUs are accepted and processed on all interfaces.

To configure BPDU protection for all edge ports for a particular spanning-tree protocol::

1. Enable edge port blocking for a particular spanning-tree protocol:

```

[edit]
user@host# set protocols (mstp | rstp | vstp) bpdv-block-on-edge

```

2. Verify BPDU protection for edge ports.

```

[edit]

```

```

protocols {
  (mstp | rstp | vstp) {
    bpdu-block-on-edge;
  }
}

```

**Related
Documentation**

- Root Protection for Spanning-Tree Instance Interfaces Overview on page 236
- BPDU Protection for Spanning-Tree Instance Interfaces Overview on page 238
- BPDU Protection on All Edge Ports of the Bridge on page 240

Checking the Status of Spanning-Tree Instance Interfaces

On an MX Series router with a spanning-tree protocol enabled, the detection of a possible bridging loop from spanning-tree protocol operation can raise a bridge protocol data unit (BPDU) error condition on the affected spanning-tree instance interface.

To check whether a spanning-tree instance interface is blocked due to a BPDU error condition:

1. To check the status of spanning-tree instance interface, use the **show interfaces** command.

```
user@host> show interfaces interface-name
```

2. You can determine the status of the interface as follows:
 - If the **BPDU Error** field is **none**, the interface is enabled.
 - If the **BPDU Error** field is **Detected** and the link is **down**, the interface is blocked.

**Related
Documentation**

- Root Protection for Spanning-Tree Instance Interfaces Overview on page 236
- BPDU Protection for Spanning-Tree Instance Interfaces Overview on page 238
- BPDU Protection for Individual Spanning-Tree Instance Interfaces on page 238
- BPDU Protection on All Edge Ports of the Bridge on page 240
- Clearing the Blocked Status of a Spanning-Tree Instance Interface on page 241

Clearing the Blocked Status of a Spanning-Tree Instance Interface

To clear the blocked status of a spanning-tree instance interface:

- Use the **clear error bpdu** operational mode command.

```
user@host> clear error bpdu interface interface-name
```



NOTE: When you configure BPDU protection on individual interfaces (as opposed to on all the edge ports of the bridge), you can use the `disable-timeout seconds` option to specify that a blocked interface is automatically cleared after the specified time interval elapses (unless the interval is 0). For configuration details, see “Configuring BPDU Protection on Individual Interfaces” on page 239.

Related Documentation

- Root Protection for Spanning-Tree Instance Interfaces Overview on page 236
- BPDU Protection for Spanning-Tree Instance Interfaces Overview on page 238
- BPDU Protection for Individual Spanning-Tree Instance Interfaces on page 238
- BPDU Protection on All Edge Ports of the Bridge on page 240
- Checking the Status of Spanning-Tree Instance Interfaces on page 241

VPLS Root Protection Topology Change Actions

The following topics describe VPLS root protection topology change actions:

- VPLS Multihomed Layer 2 Ring and MPLS Infrastructure Overview on page 242
- VPLS Multihomed Layer 2 Ring and MPLS Infrastructure Topology on page 243
- VPLS Multihoming: Priority of the Backup Bridge on page 245
- VPLS Multihoming: Hold Time Before Switching to Primary Priority on page 245
- VPLS Multihoming: System Identifier for Bridges in the Ring on page 246
- VPLS Multihoming: Bridge Flush of MAC Cache on Topology Change on page 247
- Configuring VPLS Root Protection Topology Change Actions to Control Global Spanning Tree Behavior on page 247
- Configuring VPLS Root Protection Topology Change Actions to Control VLAN Spanning Tree Behavior on page 249

VPLS Multihomed Layer 2 Ring and MPLS Infrastructure Overview

Redundancy is built into many networks through the use of alternate links and paths, which often take the shape of rings.

In the case of multiple hosts attached to customer edge (CE) routers and provider edge (PE) routers to secure virtual private LAN service (VPLS), this practice is often called *multihoming*:

- Multiple hosts attach to CE routers, which are attached to each other as well as to the PE routers that access the VPLS network cloud. Any single link between the edge routers can fail without impacting the hosts' access to the VPLS services.
- This Layer 2 ring connects to the multiprotocol link switching (MPLS) infrastructure through two PE routers. Link breaks on the ring are protected by running a version of the spanning-tree protocol with the root-protect option enabled.

The virtual private network (VPN) protocols at Layer 3, however, are not aware of the blocking state that results from this root protection setup (rings or loops are not permitted at Layer 2 because the Layer 2 protocols will not function properly).

However, to keep the Layer 2 ring functioning in a multihomed environment with link failures, the spanning-tree protocol running on the MX Series routers requires the following additional configuration:

- The VPN protocols have to act on the blocking and unblocking of interfaces by the spanning-tree protocol. Specifically, media access control (MAC) flush messages need to be sent by the blocking PE router to LDP peers in order to flush the MAC addresses learned when other interface ports were blocked.
- Also, if an active PE router with VPLS root protection bridging enabled loses VPLS connectivity, root protection requires that the bridge switch to the other PE router to maintain connectivity. The spanning-tree protocol needs to be aware of the status of the VPLS connectivity on the PE router. If the MAC address cache is not flushed when the topology changes, frames could be sent to the wrong device.

You can control the actions taken by the MX Series router when the topology changes in a multihomed Layer 2 ring VPLS environment using *VPLS root protection*.

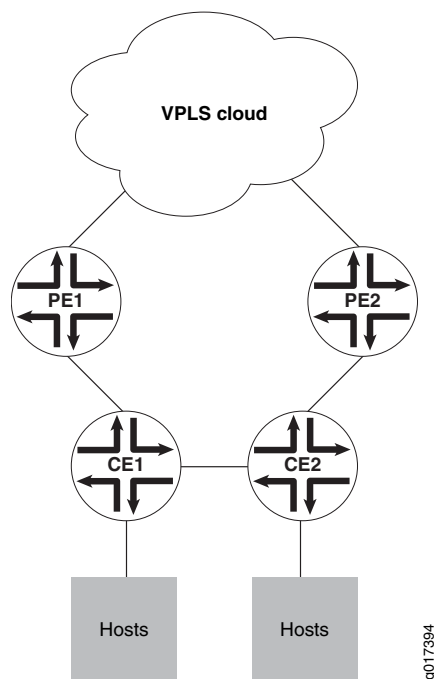
Related Documentation

- VPLS Multihomed Layer 2 Ring and MPLS Infrastructure Topology on page 243
- Configuring VPLS Root Protection Topology Change Actions to Control Global Spanning Tree Behavior on page 247
- Configuring VPLS Root Protection Topology Change Actions to Control VLAN Spanning Tree Behavior on page 249
- Example: Configuring VPLS Root Topology Change Actions on page 256

VPLS Multihomed Layer 2 Ring and MPLS Infrastructure Topology

Figure 3 on page 244 shows hosts connected to CE routers and to a VPLS network through two PE routers. The CE routers are also connected, forming a kind of ring structure.

Figure 3: VPLS Multihoming Configuration



The two PE routers have their own links to a VPLS network service, but are not directly connected to each other. All four edge routers run some type of spanning-tree protocol with root protection enabled, and only one PE interface will be in the forwarding state, the other being blocked.

Assume this forwarding interface is through PE1. If the link between CE1 and CE2 fails, then the blocking PE2 interface must detect a root protection switch and move to the forwarding state. All of the MAC addresses learned by CE2 that connect to the VPLS network service through PE1 need to be flushed. In the same way, when the link between CE1 and CE2 is restored, PE2 again detects the root protection switch and begins blocking again. Now all of the MAC addresses learned by CE2 that connect through PE2 need to be flushed. All of this is controlled by configuring VPLS root protection topology change actions on the CE routers.

Also, at a global level, each type of spanning-tree protocol will have a priority hold time associated with it. This is the number of seconds in the range from 1 through 255 seconds that the system waits to switch to the primary priority when the first core domain comes up. The default is 2 seconds. This allows the maximum number of core domains to come up, and some might be slower than others.

Related Documentation

- VPLS Multihomed Layer 2 Ring and MPLS Infrastructure Overview on page 242
- Configuring VPLS Root Protection Topology Change Actions to Control Global Spanning Tree Behavior on page 247
- Configuring VPLS Root Protection Topology Change Actions to Control VLAN Spanning Tree Behavior on page 249
- Example: Configuring VPLS Root Topology Change Actions on page 256

VPLS Multihoming: Priority of the Backup Bridge

When an MX Series router in a VPLS multihomed Layer 2 ring is running a spanning-tree protocol with root protection enabled, you can modify the default actions taken by the MX Series router when the topology changes. To do this, configure the VPLS root protection topology change actions.

The default value of the backup bridge is **32,768**. You can set the backup bridge priority to a value from **0** through **61440**, in increments of 4096.

To change the default value, you can use the following statement:

```
backup-bridge-priority vpls-ring-backup-bridge-priority
```

You can include the statement at the **[edit protocols (mstp | rstp | vstp)]** hierarchy level (to control global spanning-tree protocol behavior) or at the **[edit protocols vstp vlan *vlan-id*]** hierarchy level (to control a particular VLAN).



NOTE: VPLS root topology change actions are configured independently of VPLS, the spanning-tree protocol, or the spanning-tree protocol root protect option.

Related Documentation

- VPLS Multihomed Layer 2 Ring and MPLS Infrastructure Overview on page 242
- VPLS Multihoming: Hold Time Before Switching to Primary Priority on page 245
- Configuring VPLS Root Protection Topology Change Actions to Control Global Spanning Tree Behavior on page 247
- Example: Configuring VPLS Root Topology Change Actions on page 256

VPLS Multihoming: Hold Time Before Switching to Primary Priority

When an MX Series router in a VPLS multihomed Layer 2 ring is running a spanning-tree protocol with root protection enabled, you can modify the default actions taken by the MX Series router when the topology changes. To do this, configure the VPLS root protection topology change actions.

The default number of seconds to hold before switching to the primary priority when the first core domain comes up is 2 seconds.

To change the default value, you can use the following statement:

```
priority-hold-time seconds
```

You can include the statement at the **[edit protocols (mstp | rstp | vstp)]** hierarchy level (to control global spanning-tree protocol behavior) or at the **[edit protocols vstp vlan *vlan-id*]** hierarchy level (to control a particular VLAN).



NOTE: VPLS root topology change actions are configured independently of VPLS, the spanning-tree protocol, or the spanning-tree protocol root protect option.

Related Documentation

- VPLS Multihomed Layer 2 Ring and MPLS Infrastructure Overview on page 242
- VPLS Multihoming: Priority of the Backup Bridge on page 245
- Configuring VPLS Root Protection Topology Change Actions to Control Global Spanning Tree Behavior on page 247
- Example: Configuring VPLS Root Topology Change Actions on page 256

VPLS Multihoming: System Identifier for Bridges in the Ring

When an MX Series router in a VPLS multihomed Layer 2 ring is running a spanning-tree protocol with root protection enabled, you can modify the default actions taken by the MX Series router when the topology changes. To do this, configure the VPLS root protection topology change actions.

The system identifier for bridges in the ring is not configured by default.

To configure a system identifier for bridges in the ring, you can use the following statement:

```
system-id system-id-value bridge-host-ip-address(es)
```

You can include the statement at the **[edit protocols (mstp | rstp | vstp)]** hierarchy level (to control global spanning-tree protocol behavior) or at the **[edit protocols vstp vlan *vlan-id*]** hierarchy level (to control a particular VLAN).



NOTE: VPLS root topology change actions are configured independently of VPLS, the spanning-tree protocol, or the spanning-tree protocol root protect option.

Related Documentation

- VPLS Multihomed Layer 2 Ring and MPLS Infrastructure Overview on page 242
- VPLS Multihoming: Priority of the Backup Bridge on page 245
- VPLS Multihoming: Hold Time Before Switching to Primary Priority on page 245
- Configuring VPLS Root Protection Topology Change Actions to Control Global Spanning Tree Behavior on page 247
- Example: Configuring VPLS Root Topology Change Actions on page 256

VPLS Multihoming: Bridge Flush of MAC Cache on Topology Change

When an MX Series router in a VPLS multihomed Layer 2 ring is running a spanning-tree protocol with root protection enabled, you can modify the default actions taken by the MX Series router when the topology changes. To do this, configure the VPLS root protection topology change actions.

By default, if root protect is enabled and then the topology changes, the bridges do not flush the media access control (MAC) address cache of the MAC addresses learned when other interface ports were blocked.

To change the default behavior, you can use the following statement:

vpls-flush-on-topology-change

You can include the statement at the **[edit protocols (mstp | rstp | vstp)]** hierarchy level (to control global spanning-tree protocol behavior) or at the **[edit protocols vstp vlan *vlan-id*]** hierarchy level (to control a particular VLAN).

Specifically, MAC flush messages are sent from the blocked PE to LDP peers based on the mapping of system identifier to IP addresses as specified using the **system-id** statement.



NOTE: VPLS root topology change actions are configured independently of VPLS, the spanning-tree protocol, or the spanning-tree protocol root protect option.

Related Documentation

- VPLS Multihomed Layer 2 Ring and MPLS Infrastructure Overview on page 242
- Configuring VPLS Root Protection Topology Change Actions to Control Global Spanning Tree Behavior on page 247
- Configuring VPLS Root Protection Topology Change Actions to Control VLAN Spanning Tree Behavior on page 249
- Example: Configuring VPLS Root Topology Change Actions on page 256

Configuring VPLS Root Protection Topology Change Actions to Control Global Spanning Tree Behavior

To configure VPLS root protection topology change actions to control global spanning tree behavior:

1. Enable configuration of the spanning-tree protocol:

```
[edit]
user@host# edit protocols (mstp | rstp | vstp)
```

2. (Optional) Change the priority of the backup bridge in a VPLS multihomed Layer 2 ring with MPLS infrastructure:

```
[edit protocols (rstp | mstp | vstp)]
```

```
user@host# set backup-bridge-priority vpls-ring-backup-bridge-priority
```

3. (Optional) Change number of seconds to hold before switching to the primary priority when the first core domain comes up:

```
[edit protocols (rstp | mstp | vstp)]
user@host# set priority-hold-time seconds
```

4. Configure the system identifier for bridges in the ring:

```
[edit protocols (rstp | mstp | vstp)]
user@host# set system-id system-id-value bridge-host-ip-address(es)
```

The **system-id-value** is configured in the format **nnnnnn:nnnnnn**, where **n** = any digit from 0 to 9.

Each **bridge-host-ip-address** is a valid host IP address with a /32 mask.



NOTE: There are no default values for the system identifier or host IP addresses.

5. Configure bridges to flush the MAC address cache (of the MAC addresses learned when other interfaces ports were blocked) when the spanning-tree topology changes:

```
[edit protocols (rstp | mstp | vstp)]
user@host# setvpls-flush-on-topology-change
```

6. Verify the configuration of VPLS root protection topology change actions to control global spanning tree behavior:

```
[edit]
protocols {
  (mstp | rstp | vstp) {
    backup-bridge-priority priority; # Default is 32,768.
    priority-hold-time seconds; # Default is 2 seconds.
    system-id system-id-value {
      ip-address;
    }
    vpls-flush-on-topology-change;
  }
}
```

Related Documentation

- VPLS Multihomed Layer 2 Ring and MPLS Infrastructure Overview on page 242
- VPLS Multihomed Layer 2 Ring and MPLS Infrastructure Topology on page 243
- Configuring VPLS Root Protection Topology Change Actions to Control VLAN Spanning Tree Behavior on page 249
- Example: Configuring VPLS Root Topology Change Actions on page 256

Configuring VPLS Root Protection Topology Change Actions to Control VLAN Spanning Tree Behavior

To configure VPLS root protection topology change actions to control a particular VLAN:

1. Enable configuration of the spanning-tree protocol VLAN:

```
[edit]
user@host# edit protocols vstp vlan vlan-id
```

2. Optional) Change the priority of the backup bridge in a VPLS multihomed Layer 2 ring with MPLS infrastructure:

```
[edit protocols vstp vlan vlan-id]
user@host# set backup-bridge-priority vpls-ring-backup-bridge-priority
```

3. (Optional) Change the hold time before switching to the primary priority when the first core domain comes up:

```
[edit protocols vstp vlan vlan-id]
user@host# set priority-hold-time seconds
```

4. Configure the system identifier for bridges in the ring:

```
[edit protocols vstp vlan vlan-id]
user@host# set system-id system-id-value bridge-host-ip-address(es)
```

The ***system-id-value*** is configured in the format *nnnnnn:nnnnnn*, where *n* = any digit from 0 to 9.

Each ***bridge-host-ip-address*** is a valid host IP address with a /32 mask.



NOTE: There are no default values for the system identifier or host IP addresses.

5. Configure bridges to flush the MAC address cache (of the MAC addresses learned when other interfaces ports were blocked) when the spanning-tree topology changes:

```
[edit protocols vstp vlan vlan-id]
user@host# set vpls-flush-on-topology-change
```

6. Verify the configuration of VPLS root protection topology change actions to control a particular VLAN:

```
[edit]
protocols {
  vstp {
    vlan vlan-id {
      backup-bridge-priority priority; # Default is 32,768.
      priority-hold-time seconds; # Default is 2 seconds.
      system-id system-id-value {
        ip-address;
      }
      vpls-flush-on-topology-change;
    }
  }
}
```

- Related Documentation**
- VPLS Multihomed Layer 2 Ring and MPLS Infrastructure Overview on page 242
 - VPLS Multihomed Layer 2 Ring and MPLS Infrastructure Topology on page 243
 - Configuring VPLS Root Protection Topology Change Actions to Control Global Spanning Tree Behavior on page 247
 - Example: Configuring VPLS Root Topology Change Actions on page 256

Layer 2 Protocol Tunneling Through a Network

The following topics describe Layer 2 protocol tunneling through a network:

- Layer 2 Protocol Tunneling Through a Network Overview on page 250
- MAC Address Rewriting Enabled for Layer 2 Protocol Tunneling on page 251
- Layer 2 Protocol Tunnel Interface on page 251
- Layer 2 Protocol to be Tunneled on page 252
- Configuring Layer 2 Protocol Tunneling on page 253
- Checking for a MAC Rewrite Error Condition Blocking a Spanning-Tree Instance Interface on page 254
- Clearing a MAC Rewrite Error Condition Blocking a Spanning-Tree Instance Interface on page 254

Layer 2 Protocol Tunneling Through a Network Overview

Layer 2 protocol tunneling allows Layer 2 protocol data units (PDUs) to be tunneled through a network. This is useful to provide a single spanning-tree protocol domain for subscribers across a service provider network. It is also useful for tunneling Cisco Discovery Protocol (CDP) or VLAN Trunk Protocol (VTP) PDUs across a network.

When a control packet for STP, CDP, or VTP is received on a service provider edge port configured for Layer 2 protocol tunneling, the multicast destination MAC address is rewritten with the predefined multicast tunnel MAC address of **01:00:0c:cd:cd:d0**. The packet is transported across the provider network transparently to the other end of the tunnel and the original multicast destination MAC address is restored when the packet is transmitted.

If a packet is received on a tunnel interface that already has a destination multicast MAC address of **01:00:0c:cd:cd:d0**, the port enters an error state and is shut down. To clear the error condition, the administrator must enter the **clear error mac-rewrite interface *interface-name*** command.

Layer 2 protocol tunneling is supported on MX Series routers with enhanced queueing Dense Port Concentrators (DPCs).



NOTE: When an MX Series router sends a RADIUS access request, the **Chargeable-User-Identity** parameter is sent with an empty field. For more information about configuring RADIUS, see the *Junos Subscriber Access Configuration Guide*.

Related Documentation

- Configuring Layer 2 Protocol Tunneling on page 253
- Checking for a MAC Rewrite Error Condition Blocking a Spanning-Tree Instance Interface on page 254
- Clearing a MAC Rewrite Error Condition Blocking a Spanning-Tree Instance Interface on page 254

MAC Address Rewriting Enabled for Layer 2 Protocol Tunneling

To configure Layer 2 protocol tunneling, you must enable MAC address rewriting by installing the destination multicast tunnel MAC address of **01:00:0c:cd:cd:d0** in the MAC table.

To enable MAC address rewriting, include the **mac-rewrite** statement at the **[edit protocols layer2-control]** hierarchy level.

When enabling MAC address rewriting for Layer 2 protocol tunneling, the following guidelines apply:

- You can enable Layer 2 protocol tunneling for untagged interfaces.
- You can enable Layer 2 protocol tunneling for single-identifier tagged ports.
- You cannot enable Layer 2 protocol tunneling for double identifier tagged interfaces

Related Documentation

- Layer 2 Protocol Tunneling Through a Network Overview on page 250
- Layer 2 Protocol Tunnel Interface on page 251
- Layer 2 Protocol to be Tunneled on page 252
- Configuring Layer 2 Protocol Tunneling on page 253

Layer 2 Protocol Tunnel Interface

To configure the interface where Layer 2 protocol tunneling is enabled, include the **interface ge-fpc/pic/port** statement at the **[edit protocols layer2-control]** hierarchy level.

Keep the following guidelines in mind when configuring Layer 2 protocol tunneling:

- Layer 2 protocol tunneling is supported on MX Series routers with enhanced queueing Dense Port Concentrators (DPCs).
- Layer 2 protocol tunneling must be configured on the interfaces at each end of the tunnel.

- You can enable Layer 2 protocol tunneling for untagged interfaces and single-identifier tagged interfaces only.
- For single-identifier tagged ports, configure a logical interface with the native VLAN identifier. This configuration associates the untagged control packets with a logical interface.
- You cannot enable Layer 2 protocol tunneling for double identifier tagged interfaces.

Related Documentation

- Layer 2 Protocol Tunneling Through a Network Overview on page 250
- MAC Address Rewriting Enabled for Layer 2 Protocol Tunneling on page 251
- Layer 2 Protocol to be Tunneled on page 252
- Configuring Layer 2 Protocol Tunneling on page 253

Layer 2 Protocol to be Tunneled

To configure Layer 2 protocol tunneling, you must specify the protocol that is to be tunneled using the Layer 2 tunnel:

- **cdp**—Cisco Discovery Protocol.
- **stp**—All versions of the spanning-tree protocol.
- **vtp**—Tunnel the VLAN trunk protocol.

For each protocol specified, a static destination MAC address corresponding to the protocol being tunneled is installed in the MAC table.

To specify the protocol that will be tunneled by the Layer 2 protocol tunneling, you can include the **protocol (cdp | stp | vtp)** statement at the **[edit protocols layer2-control mac-rewrite interface ge-fpc/pic/port]** hierarchy level.



NOTE: When CDP, STP, or VTP is configured for tunneling on a customer-facing port in a provider bridge, the corresponding protocol should not be enabled for operation on that interface.

Related Documentation

- Layer 2 Protocol Tunneling Through a Network Overview on page 250
- MAC Address Rewriting Enabled for Layer 2 Protocol Tunneling on page 251
- Layer 2 Protocol Tunnel Interface on page 251
- Configuring Layer 2 Protocol Tunneling on page 253

Configuring Layer 2 Protocol Tunneling

You can configure Layer 2 protocol tunneling under the following hierarchy levels:

- **[edit]**
- **[edit routing-instances *routing-instance-name*]**

To configure Layer 2 protocol tunneling:

1. Enable MAC address rewriting for Layer 2 protocol tunneling:

```
[edit ... ]
user@host# edit protocols layer2-control mac-rewrite
```

2. Configure the Layer 2 protocol tunnel interface:

```
[edit ... protocols layer2-control mac-rewrite]
user@host# edit interface ge-fpc/pic/port
```

3. Configure the Layer 2 protocol to be tunneled:

```
[edit ... protocols layer2-control mac-rewrite interface ge-fpc/pic/port]
user@host# set protocol (cdp | stp | vtp)
```

4. Verify the configuration:

```
user@host> show ... protocols

protocols {
  layer2-control {
    mac-rewrite {
      interface ge-fpc/pic/port {
        protocol (cdp | stp | vtp);
      }
    }
  }
}
```

Related Documentation

- Layer 2 Protocol Tunneling Through a Network Overview on page 250
- Checking for a MAC Rewrite Error Condition Blocking a Spanning-Tree Instance Interface on page 254
- Clearing a MAC Rewrite Error Condition Blocking a Spanning-Tree Instance Interface on page 254

Checking for a MAC Rewrite Error Condition Blocking a Spanning-Tree Instance Interface

To check whether a spanning-tree instance interface is blocked due to a MAC rewrite error condition:

1. Use the **show interfaces** operational mode command:

```
user@host> show interfaces interface-name
```

2. You can determine the status of the interface as follows:

- If the value in the **Physical interface** includes **Enabled, Physical link is Up** and the value of the **BPDU Error** field is **None**, the interface is enabled
- If the value in the **Physical interface** field is **Enabled, Physical link is Down** and the value in the **BPDU Error** field is **Detected**, the interface is blocked.

Related Documentation

- Layer 2 Protocol Tunneling Through a Network Overview on page 250
- Configuring Layer 2 Protocol Tunneling on page 253
- Clearing a MAC Rewrite Error Condition Blocking a Spanning-Tree Instance Interface on page 254

Clearing a MAC Rewrite Error Condition Blocking a Spanning-Tree Instance Interface

To clear the blocked status of a spanning-tree instance interface:

- Use the **clear error bpdu** operational mode command.

```
user@host> clear error bpdu interface interface-name
```

Related Documentation

- Layer 2 Protocol Tunneling Through a Network Overview on page 250
- Configuring Layer 2 Protocol Tunneling on page 253
- Checking for a MAC Rewrite Error Condition Blocking a Spanning-Tree Instance Interface on page 254

Examples of Spanning-Tree Protocol Configurations

- Example: Enabling Loop Protection for Spanning-Tree Protocols on page 255
- Example: Blocking BPDUs on Aggregated Ethernet for 600 Seconds on page 255
- Example: Configuring VPLS Root Topology Change Actions on page 256
- Example: Tracing Spanning-Tree Protocol Operations on page 256

Example: Enabling Loop Protection for Spanning-Tree Protocols

This example blocks and logs the non-designated RSTP port **ge-1/2/0** after the BPDU timeout interval expires:

```
[edit]
protocols {
  rstp {
    interface ge-1/2/0 {
      bpd-timeout-action block;
    }
  }
}
```



NOTE: This is not a complete configuration. You must also fully configure RSTP, including the **ge-1/2/0** interface.

Example: Blocking BPDUs on Aggregated Ethernet for 600 Seconds

The following example, when used with a full bridge configuration with aggregated Ethernet, blocks BPDUs on aggregated interface **ae0** for ten minutes (600 seconds) before enabling the interface again:

```
[edit protocols layer2-control]
bpd-Block {
  interface ae0;
  disable-timeout 600;
}
```

- Related Documentation**
- Root Protection for Spanning-Tree Instance Interfaces Overview on page 236
 - BPDU Protection for Spanning-Tree Instance Interfaces Overview on page 238
 - BPDU Protection for Individual Spanning-Tree Instance Interfaces on page 238
 - BPDU Protection on All Edge Ports of the Bridge on page 240
 - Checking the Status of Spanning-Tree Instance Interfaces on page 241
 - Clearing the Blocked Status of a Spanning-Tree Instance Interface on page 241

Example: Configuring VPLS Root Topology Change Actions

This example configures a bridge priority of **36k**, a backup bridge priority of **44k**, a priority hold time value of **60** seconds, a system identifier of **000203:040506** for IP address **10.1.1.1/32**, and sets the bridge to flush the MAC cache on a topology change for MSTP only.

```
[edit]
protocols {
  mstp {
    bridge-priority 36k;
    backup-bridge-priority 44k;
    priority-hold-time 60;
    system-id 000203:040506 {
      10.1.1.1/32;
    }
    vpls-flush-on-topology-change;
  }
}
```



NOTE: This is not a complete configuration.

- Related Documentation**
- VPLS Multihomed Layer 2 Ring and MPLS Infrastructure Overview on page 242
 - VPLS Multihomed Layer 2 Ring and MPLS Infrastructure Topology on page 243
 - Configuring VPLS Root Protection Topology Change Actions to Control Global Spanning Tree Behavior on page 247
 - Configuring VPLS Root Protection Topology Change Actions to Control VLAN Spanning Tree Behavior on page 249

Example: Tracing Spanning-Tree Protocol Operations

Trace only unusual or abnormal operations to **/var/log/stp-log**:

```
[edit]
routing-options {
  traceoptions {
    file /var/log/routing-log;
    flag errors;
```

```
    }  
  }  
  protocols {  
    rstp {  
      traceoptions {  
        file /var/log/stp-log;  
      }  
    }  
  }  
}
```

- Related Documentation**
- [Spanning-Tree Protocols Supported on MX Series Routers](#) on page 205
 - [Tracing Spanning-Tree Operations](#) on page 230

CHAPTER 21

Summary of Spanning-Tree Protocol Configuration Statements

The following sections explain each of the Rapid Spanning-Tree Protocol (RSTP) and Multiple Spanning-Tree Protocol (MSTP) configuration statements. The statements are organized alphabetically.

backup-bridge-priority

Syntax	<code>backup-bridge-priority <i>priority</i>;</code>
Hierarchy Level	<code>[edit logical-systems <i>logical-system-name</i> protocols (mstp rstp)],</code> <code>[edit logical-systems <i>logical-system-name</i> protocols vstp vlan <i>vlan-id</i>],</code> <code>[edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> protocols</code> <code> (mstp rstp)],</code> <code>[edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> protocols</code> <code> vstp vlan <i>vlan-id</i>],</code> <code>[edit protocols (mstp rstp)],</code> <code>[edit protocols vstp vlan <i>vlan-id</i>],</code> <code>[edit routing-instances <i>routing-instance-name</i> protocols (mstp rstp)],</code> <code>[edit routing-instances <i>routing-instance-name</i> protocols vstp vlan <i>vlan-id</i>]</code>
Release Information	Statement introduced in Junos OS Release 10.0.
Description	Determine the priority of the backup bridge in a VPLS multihomed Layer 2 ring with MPLS infrastructure.
Options	<i>priority</i> —The backup bridge priority can be set only in increments of 4096. Range: 0 through 61,440 Default: 32,768
Required Privilege Level	<code>routing</code> —To view this statement in the configuration. <code>routing-control</code> —To add this statement to the configuration.
Related Documentation	<ul style="list-style-type: none">• VPLS Multihomed Layer 2 Ring and MPLS Infrastructure Overview on page 242• Configuring VPLS Root Protection Topology Change Actions to Control VLAN Spanning Tree Behavior on page 249• Configuring VPLS Root Protection Topology Change Actions to Control Global Spanning Tree Behavior on page 247• VPLS Multihoming: Priority of the Backup Bridge on page 245

bpdu-block

Syntax	bpdu-block { interface <i>interface-name</i> ; disable-timeout <i>seconds</i> ; }
Hierarchy Level	[edit protocols layer2-control]
Release Information	Statement introduced in Junos OS Release 9.4.
Description	Enable BPDU blocking on an interface. The remaining statements are described 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"> • BPDU Protection for Spanning-Tree Instance Interfaces Overview on page 238 • BPDU Protection for Individual Spanning-Tree Instance Interfaces on page 238 • Configuring BPDU Protection on Individual Interfaces on page 239

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"> • BPDU Protection for Spanning-Tree Instance Interfaces Overview on page 238 • BPDU Protection on All Edge Ports of the Bridge on page 240 • Configuring BPDU Protection on All Edge Ports on page 240

bpdu-destination-mac-address (Spanning Tree)

Syntax	<code>bpdu-destination-mac-address provider-bridge-group;</code>
Hierarchy Level	[edit logical-systems <i>logical-system-name</i> protocols (mstp rstp)], [edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> protocols (mstp rstp)], [edit protocols (mstp rstp)], [edit routing-instances <i>routing-instance-name</i> protocols (mstp rstp)]
Release Information	Statement introduced in Junos OS Release 9.2. Support for logical systems added in Junos OS Release 9.6.
Description	Enable MX Series router to participate in the provider Rapid Spanning-Tree Protocol (RSTP) instance or a provider Multiple Spanning-Tree Protocol (MSTP) instance.
Default	If the bpdu-destination-mac-address statement is not configured, the bridge participates in the customer RSTP instance, transmitting and receiving standard RSTP BPDU packets.
Options	provider-bridge-group —The destination MAC address of the BPDU packets transmitted is the provider bridge group address 01:80:c2:00:00:08 . Received BPDU packets with this destination MAC address are accepted and passed to the Routing Engine.
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">• BPDU Overview on page 206• Provider Bridge Participation in RSTP or MSTP Instances on page 211• Configuring Rapid Spanning-Tree Protocol on page 217• Configuring Multiple Spanning-Tree Protocol on page 220

bpdu-timeout-action

Syntax	<code>bpdu-timeout-action (log block);</code>
Hierarchy Level	<code>[edit ical-systems <i>ical-system-name</i> protocols (mstp rstp vstp)],</code> <code>[edit ical-systems <i>ical-system-name</i> routing-instances <i>routing-instance-name</i> protocols</code> <code> (mstp rstp vstp)],</code> <code>[edit protocols (mstp rstp vstp)],</code> <code>[edit routing-instances <i>routing-instance-name</i> protocols (mstp rstp vstp)]</code>
Release Information	<p>Statement introduced in Junos OS Release 9.4.</p> <p>Support for ical systems added in Junos OS Release 9.6.</p>
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	<p>log—The interface logs the fact that it has not received BPDUs during the timeout interval.</p> <p>block—The interface is blocked and the fact that the interface has not received BPDUs during the timeout interval is logged.</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"> • Loop Protection for Spanning-Tree Instance Interfaces Overview on page 233 • Configuring Loop Protection for a Spanning-Tree Instance Interface on page 235 • Example: Enabling Loop Protection for Spanning-Tree Protocols on page 255

bridge-priority

Syntax	<code>bridge-priority <i>priority</i>;</code>
Hierarchy Level	<code>[edit logical-systems <i>logical-system-name</i> protocols (mstp rstp)],</code> <code>[edit logical-systems <i>logical-system-name</i> protocols mstp msti <i>msti-id</i>],</code> <code>[edit logical-systems <i>logical-system-name</i> protocols vstp vlan <i>vlan-id</i>],</code> <code>[edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> protocols</code> <code> (mstp rstp)],</code> <code>[edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i></code> <code> protocols mstp msti <i>msti-id</i>],</code> <code>[edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> protocols</code> <code> vstp vlan <i>vlan-id</i>],</code> <code>[edit protocols (mstp rstp)],</code> <code>[edit protocols mstp msti <i>msti-id</i>],</code> <code>[edit protocols vstp vlan <i>vlan-id</i>],</code> <code>[edit routing-instances <i>routing-instance-name</i> protocols (mstp rstp)],</code> <code>[edit routing-instances <i>routing-instance-name</i> protocols mstp msti <i>msti-id</i>],</code> <code>[edit routing-instances <i>routing-instance-name</i> protocols vstp vlan <i>vlan-id</i>]</code>
Release Information	Statement introduced in Junos OS Release 8.4. Support for logical systems added in Junos OS Release 9.6.
Description	Determine which bridge is elected as the root bridge. If two bridges have the same path cost to the root bridge, the bridge priority determines which bridge becomes the designated bridge for a LAN segment.
Options	<i>priority</i> —The bridge priority can be set only in increments of 4096. Range: 0 through 61,440 Default: 32,768
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">Bridge Priority for Election of Root Bridge and Designated Bridge on page 212

configuration-name

Syntax	configuration-name <i>configuration-name</i> ;
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. Support for logical systems added in Junos OS Release 9.6.
Description	The configuration name 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">• BPDUs Overview on page 206• Configuring Multiple Spanning-Tree Protocol on page 220

cost

Syntax	<code>cost cost;</code>
Hierarchy Level	<pre>[edit logical-systems <i>logical-system-name</i> protocols (mstp rstp vstp) interface <i>interface-name</i>], [edit logical-systems <i>logical-system-name</i> protocols mstp msti <i>msti-id</i> interface <i>interface-name</i>], [edit logical-systems <i>logical-system-name</i> protocols vstp vlan <i>vlan-id</i> interface <i>interface-name</i>], [edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> protocols (mstp rstp vstp) interface <i>interface-name</i>], [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>], [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>], [edit protocols (mstp rstp vstp) interface <i>interface-name</i>], [edit protocols mstp msti <i>msti-id</i> interface <i>interface-name</i>], [edit protocols vstp vlan <i>vlan-id</i> interface <i>interface-name</i>], [edit routing-instances <i>routing-instance-name</i> protocols (mstp rstp vstp) interface <i>interface-name</i>], [edit routing-instances <i>routing-instance-name</i> protocols mstp msti <i>msti-id</i> interface <i>interface-name</i>], [edit routing-instances <i>routing-instance-name</i> protocols vstp vlan <i>vlan-id</i> interface <i>interface-name</i>]</pre>
Release Information	<p>Statement introduced in Junos OS Release 8.4.</p> <p>Support for logical systems added in Junos OS Release 9.6.</p>
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"> Spanning-Tree Instance Interface on page 213 Spanning-Tree Instance Interface Cost on page 214

disable

Syntax	disable;
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 9.1. Support for logical systems added in Junos OS Release 9.6.
Description	Disable the entire MSTP instance.
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 Multiple Spanning-Tree Protocol on page 220 Disabling MSTP on page 225

disable-timeout

Syntax	disable-timeout <i>seconds</i> ;
Hierarchy Level	[edit protocols layer2-control bpdu-block]
Release Information	Statement introduced in Junos OS Release 9.4.
Description	Configure the timeout value to periodically check to see if an interface is still disabled with BPDU blocking. If this option is not configured, the interface is not periodically checked and remains disabled.
Options	<p><i>seconds</i>—Disable timeout value.</p> <p>Range: 10 through 3600</p> <p>Default: If this option is not configured, the interface is not periodically checked and remains disabled.</p>
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"> BPDU Protection for Spanning-Tree Instance Interfaces Overview on page 238 BPDU Protection for Individual Spanning-Tree Instance Interfaces on page 238

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>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"> Spanning-Tree Instance Interface on page 213 Spanning-Tree Instance Interface Configured as an Edge Port on page 215

extended-system-id

Syntax	<code>extended-system-id <i>identifier</i>;</code>
Hierarchy Level	[edit logical-systems <i>logical-system-name</i> protocols rstp], [edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> protocols rstp], [edit protocols rstp], [edit routing-instances <i>routing-instance-name</i> protocols rstp]
Release Information	Statement introduced in Junos OS Release 8.3. Support for logical systems added in Junos OS Release 9.6.
Description	The extended system ID is used to specify different bridge identifiers for different RSTP or STP routing instances.
Options	<i>identifier</i> —Specify the system identifier to use for the RSTP or STP instance. Range: 0 through 4095
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"> System Identifier for Bridges in STP or RSTP Instances on page 212 Configuring Rapid Spanning-Tree Protocol on page 217

force-version

Syntax	<code>force-version stp;</code>
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. Support for logical systems added in Junos OS Release 9.6.
Description	Force the spanning-tree version to be the original IEEE 803.1D 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"> Spanning-Tree Protocols Supported on MX Series Routers on page 205 RSTP or VSTP Forced to Run as IEEE 802.1D STP on page 210 Reverting RSTP or VSTP Back From Forced IEEE 802.1D STP on page 210

forward-delay

Syntax	<code>forward-delay seconds;</code>
Hierarchy Level	<code>[edit logical-systems <i>logical-system-name</i> protocols (mstp rstp)],</code> <code>[edit logical-systems <i>logical-system-name</i> protocols vstp vlan <i>vlan-id</i>],</code> <code>[edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> protocols</code> <code> (mstp rstp)],</code> <code>[edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> protocols</code> <code> vstp vlan <i>vlan-id</i>],</code> <code>[edit protocols (mstp rstp)],</code> <code>[edit protocols vstp vlan <i>vlan-id</i>],</code> <code>[edit routing-instances <i>routing-instance-name</i> protocols (mstp rstp)],</code> <code>[edit routing-instances <i>routing-instance-name</i> protocols vstp vlan <i>vlan-id</i>]</code>
Release Information	Statement introduced in Junos OS Release 8.4. Support for logical systems added in Junos OS Release 9.6.
Description	Specify the length of time an STP bridge port remains in the listening and learning states before transitioning to the forwarding state.
Options	seconds —(Optional) Number of seconds the bridge port remains in the listening and learning states. Range: 4 through 30 Default: 15 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">Forward Delay Before Ports Transition to Forwarding State on page 213

hello-time

Syntax	hello-time <i>seconds</i> ;
Hierarchy Level	[edit logical-systems <i>logical-system-name</i> protocols (mstp rstp)], [edit logical-systems <i>logical-system-name</i> protocols vstp vlan <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 vlan <i>vlan-id</i>], [edit protocols (mstp rstp)], [edit protocols vstp vlan <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 vlan <i>vlan-id</i>]
Release Information	Statement introduced in Junos OS Release 8.4. Support for logical systems added in Junos OS Release 9.6.
Description	Specify the number of seconds between transmissions of configuration BPDUs by the root bridge.
Options	seconds —(Optional) Number of seconds between transmissions of configuration BPDUs. Range: 1 through 10 Default: 2 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"> Hello Time for Root Bridge to Transmit Hello BPDUs on page 212

interface

See the following sections:

- [interface \(BPDU Blocking\) on page 272](#)
- [interface \(Layer 2 Protocol Tunneling\) on page 272](#)
- [interface \(Spanning Tree\) on page 273](#)

interface (BPDU Blocking)

Syntax	<code>interface <i>interface-name</i>;</code>
Hierarchy Level	<code>[edit protocols layer2-control bpdu-block]</code>
Release Information	Statement introduced in Junos OS Release 9.4.
Description	Configure the interface to participate in BPDU blocking.
Options	<i>interface-name</i> —Name of a Gigabit Ethernet or 10-Gigabit Ethernet 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">• BPDU Protection for Spanning-Tree Instance Interfaces Overview on page 238• BPDU Protection for Individual Spanning-Tree Instance Interfaces on page 238• Configuring BPDU Protection on Individual Interfaces on page 239

interface (Layer 2 Protocol Tunneling)

Syntax	<pre>interface <i>interface-name</i> { protocol (cdp stp vtp); }</pre>
Hierarchy Level	<code>[edit protocols layer2-control mac-rewrite]</code>
Release Information	Statement introduced in Junos OS Release 9.1.
Description	Configure an interface for Layer 2 protocol tunneling. The remaining statement is described 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">• Layer 2 Protocol Tunneling Through a Network Overview on page 250

interface (Spanning Tree)

Syntax	<pre> interface <i>interface-name</i> { bpdutimeout-action { alarm; block; } cost <i>cost</i>; edge; mode (p2p shared); no-root-port; priority <i>interface-priority</i>; } </pre>
Hierarchy Level	<pre> [edit logical-systems <i>logical-system-name</i> protocols (mstp rstp vstp)], [edit logical-systems <i>logical-system-name</i> protocols vstp vlan <i>vlan-id</i>], [edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> protocols (mstp rstp vstp)], [edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> protocols vstp vlan <i>vlan-id</i>], [edit protocols (mstp rstp vstp)], [edit protocols vstp vlan <i>vlan-id</i>], [edit routing-instances <i>routing-instance-name</i> protocols (mstp rstp vstp)], [edit routing-instances <i>routing-instance-name</i> protocols vstp vlan <i>vlan-id</i>] </pre>
Release Information	<p>Statement introduced in Junos OS Release 8.4.</p> <p>Support for logical systems added in Junos OS Release 9.6.</p>
Description	Configure the interface to participate in the RSTP or MSTP instance.
Options	<p><i>interface-name</i>—Name of a Gigabit Ethernet or 10-Gigabit Ethernet interface.</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>
Related Documentation	<ul style="list-style-type: none"> Spanning-Tree Instance Interface on page 213

layer2-control

Syntax

```
layer2-control {  
  bpd-block {  
    interface interface-name;  
    disable-timeout seconds;  
  }  
  mac-rewrite {  
    interface interface-name {  
      protocol (cdp | stp | vtp);  
    }  
  }  
  nonstop-bridging;  
}
```

Hierarchy Level [edit protocols]

Release Information Statement introduced in Junos OS Release 8.4.
bpd-block statement added in Junos OS Release 9.4.

Description Configure Layer 2 control protocols to enable features such as Layer 2 protocol tunneling or nonstop bridging.

The remaining statements are described separately.



NOTE: For a detailed description of configuring the nonstop-bridging statement, see the [Junos OS High Availability Configuration Guide](#). When this statement is configured on routing platforms with two Routing Engines, a master Routing Engine switches over gracefully to a backup Routing Engine and preserves Layer 2 Control Protocol (L2CP) information.

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

Related Documentation

- Layer 2 Protocol Tunneling Through a Network Overview on page 250
- Layer 2 Protocol Tunnel Interface on page 251
- Layer 2 Protocol to be Tunneled on page 252
- Configuring Layer 2 Protocol Tunneling on page 253
- instance-type

mac-rewrite

Syntax	<pre>mac-rewrite { interface <i>interface-name</i> { protocol (cdp stp vtp); } }</pre>
Hierarchy Level	[edit protocols layer2-control]
Release Information	Statement introduced in Junos OS Release 9.1.
Description	<p>Enable rewriting of the MAC address for Layer 2 protocol tunneling.</p> <p>The remaining statements are described 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"> Layer 2 Protocol Tunneling Through a Network Overview on page 250

max-age

Syntax	<code>max-age <i>seconds</i>;</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 vlan <i>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 vlan <i>vlan-id</i>],</p> <p>[edit protocols (mstp rstp)],</p> <p>[edit protocols vstp vlan <i>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 vlan <i>vlan-id</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>
Description	Specify the maximum expected arrival time of hello BPDUs.
Options	<p><i>seconds</i>—(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"> Maximum Age for Awaiting Arrival of Hello BPDUs on page 212

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. 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">Configuring Multiple Spanning-Tree Protocol on page 220

mode

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>Support for logical systems added in Junos OS Release 9.6.</p>
Description	Configure link mode to identify point-to-point links.
Default	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"> Spanning-Tree Instance Interface on page 213 Spanning-Tree Instance Interface Point-to-Point Link Mode on page 215

msti

Syntax	<pre>msti <i>msti-id</i> { bridge-priority <i>priority</i>; vlan <i>vlan-id</i>; interface <i>interface-name</i> { cost <i>cost</i>; edge; priority <i>interface-priority</i>; } }</pre>
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. Support for logical systems added in Junos OS Release 9.6.
Description	Configure the Multiple Spanning Tree Protocol (MSTI) instance identifier.
Options	<p><i>msti-id</i>—MSTI instance identifier.</p> <p>Range: 1 through 64</p> <p>The remaining statements are explained separately.</p>
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 Multiple Spanning-Tree Protocol on page 220Configuring MST Instances on a Physical Interface on page 224

mstp

Syntax	<pre> mstp { bpdn-block-on-edge; bridge-priority <i>priority</i>; configuration-name <i>configuration-name</i>; disable; forward-delay <i>seconds</i>; hello-time <i>seconds</i>; max-age <i>seconds</i>; max-hops <i>hops</i>; priority-hold-time <i>seconds</i>; revision-level <i>revision-level</i>; interface <i>interface-name</i> { bpdn-timeout-action { alarm; block; } cost <i>cost</i>; edge; mode (p2p shared); no-root-port; priority <i>interface-priority</i>; } msti <i>msti-id</i> { bridge-priority <i>priority</i>; interface <i>interface-name</i> { cost <i>cost</i>; edge; priority <i>interface-priority</i>; } vlan <i>vlan-id</i>; } 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], [edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> protocols], [edit protocols], [edit routing-instances <i>routing-instance-name</i> protocols]</p>
Release Information	<p>Statement introduced in Junos OS Release 8.4. bpdn-block-on-edge statement added in Junos OS Release 9.4. bpdn-timeout-action statement added in Junos OS Release 9.4. Support for logical systems added in Junos OS Release 9.6.</p>
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

- Configuring Multiple Spanning-Tree Protocol on page 220

no-root-port

Syntax no-root-port;

Hierarchy Level [edit logical-systems *logical-system-name* protocols (mstp | rstp | vstp) interface *interface-name*],
[edit logical-systems *logical-system-name* protocols vstp vlan *vlan-id* interface *interface-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols (mstp | rstp | vstp) interface *interface-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols vstp vlan *vlan-id* interface *interface-name*],
[edit protocols (mstp | rstp | vstp) interface *interface-name*],
[edit protocols vstp vlan *vlan-id* interface *interface-name*],
[edit routing-instances *routing-instance-name* protocols (mstp | rstp | vstp) interface *interface-name*],
[edit routing-instances *routing-instance-name* protocols vstp vlan *vlan-id* interface *interface-name*]

Release Information Statement introduced in Junos OS Release 9.1.
Support for logical systems added in Junos OS Release 9.6.

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

- Root Protection for Spanning-Tree Instance Interfaces Overview on page 236
- Root Protect for a Spanning-Tree Instance Interface on page 236
- Enabling Root Protect for a Spanning-Tree Instance Interface on page 237

priority

Syntax	<code>priority interface-priority;</code>
Hierarchy Level	<pre>[edit logical-systems <i>logical-system-name</i> protocols (mstp rstp vstp) interface <i>interface-name</i>], [edit logical-systems <i>logical-system-name</i> protocols mstp msti <i>msti-id</i> interface <i>interface-name</i>], [edit logical-systems <i>logical-system-name</i> protocols vstp vlan <i>vlan-id</i> interface <i>interface-name</i>], [edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> protocols (mstp rstp vstp) interface <i>interface-name</i>], [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>], [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>], [edit protocols (mstp rstp vstp) interface <i>interface-name</i>], [edit protocols mstp msti <i>msti-id</i> interface <i>interface-name</i>], [edit protocols vstp vlan <i>vlan-id</i> interface <i>interface-name</i>], [edit routing-instances <i>routing-instance-name</i> protocols (mstp rstp vstp) interface <i>interface-name</i>], [edit routing-instances <i>routing-instance-name</i> protocols mstp msti <i>msti-id</i> interface <i>interface-name</i>], [edit routing-instances <i>routing-instance-name</i> protocols vstp vlan <i>vlan-id</i> interface <i>interface-name</i>]</pre>
Release Information	<p>Statement introduced in Junos OS Release 8.4.</p> <p>Support for logical systems added in Junos OS Release 9.6.</p>
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"> Spanning-Tree Instance Interface on page 213 Spanning-Tree Instance Interface Configured as an Edge Port on page 215 Spanning-Tree Instance Interface Priority on page 214

priority-hold-time

Syntax	<code>priority-hold-time seconds;</code>
Hierarchy Level	[edit logical-systems <i>logical-system-name</i> protocols (mstp rstp)], [edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> protocols (mstp rstp)], [edit protocols (mstp rstp)], [edit routing-instances <i>routing-instance-name</i> protocols (mstp rstp)],
Release Information	Statement introduced in Junos OS Release 10.0.
Description	Specify the number of seconds to hold before switching to the primary priority when the first core domain comes up.
Options	seconds —Number of seconds to hold before switching to primary priority. Range: 1 through 255 Default: 2 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">• VPLS Multihoming: Hold Time Before Switching to Primary Priority on page 245

protocol

Syntax	<code>protocol (cdp stp vtp);</code>
Hierarchy Level	[edit protocols layer2-control mac-rewrite interface <i>interface-name</i>],
Release Information	Statement introduced in Junos OS Release 9.1.
Description	Configure the protocol to be tunneled on an interface for Layer 2 protocol tunneling. To tunnel multiple protocols, include multiple protocol statements.
Options	cdp —Tunnel the Cisco discovery protocol. stp —Tunnel all versions of the spanning-tree protocol. vtp —Tunnel the VLAN trunk protocol.
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">• Layer 2 Protocol Tunneling Through a Network Overview on page 250• Layer 2 Protocol Tunnel Interface on page 251• Layer 2 Protocol to be Tunneled on page 252• Configuring Layer 2 Protocol Tunneling on page 253

protocols

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. 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">• Configuring Rapid Spanning-Tree Protocol on page 217• Configuring Multiple Spanning-Tree Protocol on page 220• Configuring VLAN Spanning-Tree Protocol on page 226

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. 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">Configuring Multiple Spanning-Tree Protocol on page 220

rstp

Syntax	<pre>rstp { bpd-block-on-edge; bpd-destination-mac-address provider-bridge-group; bridge-priority <i>priority</i>; extended-system-id; force-version stp; forward-delay <i>seconds</i>; hello-time <i>seconds</i>; max-age <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>; } 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 logical-systems <i>logical-system-name</i> protocols], [edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> protocols], [edit protocols], [edit routing-instances <i>routing-instance-name</i> protocols]
Release Information	Statement introduced in Junos OS Release 8.4. bpd-block-on-edge statement added in Junos OS Release 9.4. bpd-timeout-action statement added in Junos OS Release 9.4. Support for logic systems added in Junos OS Release 9.6.
Description	Configure RSTP 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">Configuring Rapid Spanning-Tree Protocol on page 217

system-id

Syntax	<code>system-id system-id-value { ip-address(es); }</code>
Hierarchy Level	[edit logical-systems <i>logical-system-name</i> protocols (mstp rstp)], [edit logical-systems <i>logical-system-name</i> protocols vstp vlan <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 vlan <i>vlan-id</i>], [edit protocols (mstp rstp)], [edit protocols vstp vlan <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 vlan <i>vlan-id</i>]
Release Information	Statement introduced in Junos OS Release 10.0.
Description	Determine the system identifier value for bridges in a VPLS multihomed Layer 2 ring with MPLS infrastructure.
Options	<p><i>system-id-value</i>—System identifier in the format <i>nnnnnn:nnnnnn</i> where <i>n</i> = any digit from 0 through 9.</p> <p>Range: Any valid value</p> <p>Default: None</p> <p><i>ip-address(es)</i>—Valid IP host addresses in the format <i>ip-address/32</i>.</p> <p>Range: Any valid IP address</p> <p>Default: None</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"> • VPLS Multihomed Layer 2 Ring and MPLS Infrastructure Overview on page 242 • VPLS Multihomed Layer 2 Ring and MPLS Infrastructure Topology on page 243 • VPLS Multihoming: System Identifier for Bridges in the Ring on page 246

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	[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 8.4. Support for logical systems added in Junos OS Release 9.6.
Description	Set STP protocol-level tracing options.
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 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	routing—To view this statement in the configuration.
Level	routing-control—To add this statement to the configuration.

Related Documentation	<ul style="list-style-type: none">• Spanning-Tree Protocol Trace Options on page 216• Tracing Spanning-Tree Operations on page 230• Example: Tracing Spanning-Tree Protocol Operations on page 256
------------------------------	--

vlan

See the following sections:

- [vlan \(MSTP\) on page 291](#)
- [vlan \(VSTP\) on page 292](#)

vlan (MSTP)

Syntax	<code>vlan <i>vlan-id</i>;</code>
Hierarchy Level	[edit logical-systems <i>logical-system-name</i> protocols mstp msti <i>msti-id</i>], [edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> protocols mstp msti <i>msti-id</i>], [edit protocols mstp msti <i>msti-id</i>], [edit routing-instances <i>routing-instance-name</i> protocols mstp msti <i>msti-id</i>]
Release Information	Statement introduced in Junos OS Release 8.4. Support for logical systems added in Junos OS Release 9.6.
Description	Configure the VLAN of an MSTI or VSTP instance or configure the VLAN range of an MSTI instance.
Options	<i>vlan-id</i> —The VLAN identifier associated with the MSTI. <i>vlan-id-range</i> —Range of VLAN identifiers associated with the MSTI in the form <i>minimum-vlan-id-maximum-vlan-id</i> . VLAN identifier ranges are not supported for VSTP. Range: 1 through 4096
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 Multiple Spanning-Tree Protocol on page 220

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. 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">Configuring VLAN Spanning-Tree Protocol on page 226

vpls-flush-on-topology-change

Syntax	vpls-flush-on-topology-change;
Hierarchy Level	[edit logical-systems <i>logical-system-name</i> protocols (mstp rstp)], [edit logical-systems <i>logical-system-name</i> protocols vstp vlan <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 vlan <i>vlan-id</i>], [edit protocols (mstp rstp)], [edit protocols vstp vlan <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 vlan <i>vlan-id</i>]
Release Information	Statement introduced in Junos OS Release 10.0.
Description	Determine the action the bridge should take when the topology of a multihomed Layer 2 ring with MPLS infrastructure changes: flush the media access control (MAC) cache or not. By default, the bridge does not flush the cache when the topology changes.
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"> VPLS Multihoming: Bridge Flush of MAC Cache on Topology Change on page 247

vstp

Syntax	<pre> vstp { bpd-block-on-edge; force-version stp; 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>; } priority-hold-time <i>seconds</i>; 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> { bpd-timeout-action { alarm; block; } cost <i>cost</i>; edge; mode (p2p shared); no-root-port; priority <i>interface-priority</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	<p>[edit logical-systems <i>logical-system-name</i> protocols], [edit logical-systems <i>logical-system-name</i> routing-instances <i>routing-instance-name</i> protocols], [edit protocols], [edit routing-instances <i>routing-instance-name</i> protocols]</p>
Release Information	<p>Statement introduced in Junos OS Release 9.0.</p> <p>bpd-block-on-edge statement added in Junos OS Release 9.4.</p> <p>bpd-timeout-action statement added in Junos OS Release 9.4.</p> <p>Support for logical systems added in Junos OS Release 9.6.</p>
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	<ul style="list-style-type: none">• Configuring VLAN Spanning-Tree Protocol on page 226

PART 7

Indexes

- Index on page 299
- Index of Statements and Commands on page 309

Index

Symbols

#, comments in configuration statements.....	xxvi
(), in syntax descriptions.....	xxvi
< >, in syntax descriptions.....	xxv
[], in configuration statements.....	xxvi
{ }, in configuration statements.....	xxvi
(pipe), in syntax descriptions.....	xxvi

A

address learning, Layer 2	
in logical systems	
configuring.....	27
advertisement-interval statement.....	37
all-failures (tracing flag)	
spanning-tree protocols.....	288

B

backup-bridge-priority statement.....	260
usage guidelines.....	247
bandwidth statement.....	175
usage guidelines.....	172
BPDU	
overview.....	206
bpdu (tracing flag).....	288
BPDU blocking for spanning-tree protocols	
Layer 2 control	
disable-timeout statement.....	267
interface (BPDU Blocking)	
statement.....	272
BPDU protection	
spanning-tree instance interface	
checking the status.....	241
clearing the blocked status.....	241
BPDU protection for spanning-tree protocols	
on all edge ports.....	240
configuration guidelines.....	240
on individual interfaces	
configuration guidelines.....	238
on individual ports.....	239
overview.....	238

bpdu-block	
usage guidelines.....	238
bpdu-block statement.....	261
bpdu-block-on-edge	
usage guidelines.....	240
bpdu-block-on-edge statement.....	261
bpdu-destination-mac-address statement	
MVRP.....	71
spanning tree.....	262
bpdu-timeout-action statement.....	263
usage guidelines.....	235
braces, in configuration statements.....	xxvi
brackets	
angle, in syntax descriptions.....	xxv
square, in configuration statements.....	xxvi
bridge domain	
dual VLAN tags.....	191
routing interface.....	185
VLAN identifier.....	189
VLAN identifier list.....	151, 152
bridge domains	
logical systems for	
configuring.....	27
example.....	28
overview.....	27
Bridge Protocol Data Unit See BPDU	
bridge-detection-state-machine (tracing	
flag).....	288
bridge-domains statement.....	176
bridge-options statement.....	177
bridge-priority statement.....	264
bridging loop error condition	
spanning-tree instance interface	
checking the status.....	241
clearing the blocked status.....	241

C**CDP**

Layer 2 Protocol Tunneling	
overview.....	250
Layer 2 protocol tunneling	
configuring.....	252
Cisco Discovery Protocol See CDP	
clear spanning-tree protocol-migration	
command.....	210
comments, in configuration statements.....	xxvi
configuration-name statement.....	265
configuring	
LLDP.....	32
conventions	
text and syntax.....	xxv
cost statement.....	266
curly braces, in configuration statements.....	xxvi
customer support.....	xxvi
contacting JTAC.....	xxvi

D**Dense Port Concentrator See DPC**

disable statement	
LLDP.....	38
mstp.....	267
usage guidelines.....	225
disable-timeout	
usage guidelines.....	238
disable-timeout statement.....	267
documentation	
comments on.....	xxvi
domain-type statement.....	178
DPC	
description.....	5
in an MX Series router	
binding to a named instance of Layer 2 port	
mirroring.....	108
displaying the number and types	
installed.....	107

E

edge port	
configuration guidelines.....	215
edge statement.....	268
error (tracing flag)	
MVRP.....	80
Ethernet	
frame counters and statistics	
in MX Series routers.....	10

Ethernet link aggregation

and load balancing	
configuring.....	25
example.....	26
overview.....	25
events (tracing flag)	
MVRP.....	80
spanning-tree protocols.....	288
example	
LLDP.....	35
example of next-hop groups	
and Layer 2 port mirroring.....	140
example of port mirroring, Layer 2	
next-hop groups.....	140
extended-system-id statement.....	269

F

firewall filter for Layer 2 port mirroring	
defining.....	113
firewall filter-driven port mirroring, Layer 2	
application on PE routers.....	96
applying to a bridge domain.....	120
applying to a logical interface.....	117
applying to a VPLS routing instance.....	122
for an aggregated Ethernet interface.....	98
Flexible Port Concentrator See FPC	
font conventions.....	xxv
force-version statement.....	269
forward-delay statement.....	270
FPC	
description.....	5
in an MX Series router	
binding to a named instance of Layer 2 port	
mirroring.....	108
displaying the number and types	
installed.....	107
frames	
Ethernet counters and statistics	
in MX Series routers.....	10

G

global instance of Layer 2 port mirroring	
configuring.....	101
disabling.....	112
displaying.....	111
global-mac-limit statement.....	199
usage guidelines.....	197
global-mac-statistics statement.....	200
usage guidelines.....	196

- global-mac-table-aging-time statement.....200
 - usage guidelines.....196
- global-no-mac-learning statement.....201
 - usage guidelines.....198
- H**
- hardware components
 - Dense Port Concentrator (DPC).....5
- hello-time statement.....271
- hold-multiplier statement.....38
- I**
- icons defined, notice.....xxiv
- IGMP snooping.....8
- interface
 - Layer 2 protocol tunnel interface
 - configuration guidelines.....251
- interface (BPDU Blocking)
 - usage guidelines for individual interfaces.....238
- interface statement
 - BPDU blocking.....272
 - bridge domain.....178
 - Layer 2 protocol tunneling.....272
 - LLDP.....39
 - MVRP.....72
 - spanning tree.....273
 - STP
 - usage guidelines.....213
 - virtual switch.....178
- interface-mac-limit statement.....179
 - set of bridge domains
 - usage guidelines.....171
 - trunk port
 - usage guidelines.....171
 - usage guidelines.....168
- irb
 - and mirroring.....182
- J**
- join-timer statement
 - MVRP.....73
- L**
- l2-learning statement.....201
- Layer 2
 - Trio parity.....7
- layer2-control routing instance
 - minimum configuration.....18
- layer2-control routing instances.....18
- layer2-control statement.....274
- Layer 2 and Layer 3 services
 - on MX Series routers.....8
- Layer 2 control
 - BPDU blocking for spanning-tree protocols
 - disable-timeout statement.....267
 - interface (BPDU Blocking)
 - statement.....272
- Layer 2 learning and forwarding
 - in logical systems
 - configuration guidelines.....27
 - configuring.....27
 - example.....28
- Layer 2 protocol tunneling
 - configuring.....253
 - enabling MAC address rewriting
 - configuration guidelines.....251
 - overview.....250
 - physical interface
 - configuration guidelines.....251
 - protocol to be tunneled
 - configuration guidelines.....252
 - spanning-tree instance interface
 - checking for a MAC rewrite error
 - condition.....254
 - clearing a MAC rewrite error
 - condition.....254
- Layer 2 VPNs.....9
- leave-timer statement
 - MVRP.....75
- leaveall-timer statement
 - MVRP.....74
- link layer discovery protocol See LLDP
- LLDP
 - configuration statements.....44
 - configuring.....32
 - example.....35
 - overview.....31
 - tracing operations.....34
- lldp statement
 - LLDP.....40
- lldp-configuration-notification-interval
 - statement.....41
- load balancing
 - and Ethernet link aggregation
 - configuring.....25
 - example.....26
 - overview.....25

logical systems	
Layer 2 learning and forwarding	
configuring.....	27
example.....	28
overview.....	27
spanning-tree protocols.....	209
loop-free topology.....	205

M

MAC address	
clearing rewrite error conditions.....	254
clearing the blocked status.....	254
enabling rewriting for Layer 2 protocol tunneling	
configuration guidelines.....	251
rewriting for Layer 2 protocol tunneling	
configuring.....	253
mac-rewrite statement.....	275
mac-statistics statement.....	180
set of bridge domains	
usage guidelines.....	172
trunk port	
usage guidelines.....	172
usage guidelines.....	170
mac-table-size statement.....	181
set of bridge domains	
usage guidelines.....	172
trunk port	
usage guidelines.....	172
usage guidelines.....	168
manuals	
comments on.....	xxvi
max-age statement.....	275
max-hops statement.....	276
mirroring	
and irb.....	182
mode statement.....	277
msti statement.....	278
MSTP	
BPDU overview.....	206
configuring.....	220
nonstop bridging support.....	206
overview.....	205
VLAN.....	291
mstp	
disabling.....	267
MSTP configuration	
BPDU destination MAC address.....	211
bridge priority.....	212

instance interface	
edge port.....	215
interface participating an the instance.....	213
interval for root bridge sending configuration	
BPDUs.....	212
link cost for determining designated bridge and	
port.....	214
link mode to identify point-to-point links.....	215
maximum interval between arrival of hello	
BPDUs.....	212
priority of interface to become root port.....	214
root protect option	
configuration guidelines.....	236
configuring.....	237
overview.....	236
time bridge port remains in listening, learning	
state.....	213
MSTI	
bridge priority.....	212
edge port.....	215
link cost for determining designated bridge	
and port.....	214
priority of interface to become root	
port.....	214
mstp statement.....	279
usage guidelines.....	220
multicast snooping.....	8
and spanning-tree protocols	
configuring.....	23
example.....	24
and VPLS root protection	
overview.....	21
multihomed environment See VPLS multihomed	
Layer 2 ring	
VPLS Layer 2 ring and multicast snooping	
configuring.....	23
example.....	24
overview.....	21
VPLS multihomed Layer 2 ring	
configuring VSTP.....	226
topology.....	243
Multiple Spanning-Tree Protocol See MSTP	
mvrp statement.....	76

N

named instances of Layer 2 port mirroring	
defining.....	104
disabling.....	112
displaying.....	111

next-hop groups
 for Layer 2 port mirroring to multiple destinations
 defining.....124, 125
 example.....140
 overview.....92
 no-dynamic-vlan statement
 MVRP.....77
 no-irb-layer-2-copy statement.....182
 no-mac-learning statement.....183
 set of bridge domains
 usage guidelines.....170
 trunk port
 usage guidelines.....170
 usage guidelines.....166
 no-root-port statement.....280
 notice icons defined.....xxiv

P

Packet Forwarding Engine *See* PIC
 binding associated ports to Layer 2
 port-mirroring.....110
 Packet Forwarding Engines
 displaying chassis information.....107
 packet-action statement.....184
 parentheses, in syntax descriptions.....xxvi
 pdu (tracing flag)
 MVRP.....80
 PIC
 binding associated ports to Layer 2
 port-mirroring.....110
 description.....5
 PIM snooping.....8
 point-to-point links.....215
 point-to-point statement
 MVRP.....78
 port mirroring, Layer 2
 example
 family ccc.....135
 family ccc with aggregated Ethernet
 links.....138
 L2VPN.....135
 L2VPN with aggregated Ethernet
 links.....138
 logical interface.....133
 multiple instances.....127
 firewall filter.....90
 applying to a bridge domain.....120
 applying to a logical interface.....117

applying to a VPLS routing instance.....122
 compared with other port mirroring
 sets.....94
 defining.....113
 example with a logical interface.....133
 example with multiple instances.....129
 overview.....90
 See also global instance
 See also named instances
 for a bridge domain
 overview of firewall filters.....90
 traffic forwarded or flooded to.....120
 for a logical interface.....117
 applying a firewall filter.....117
 defining a firewall filter.....113
 defining next-hop groups.....124
 displaying next-hop groups.....125
 example.....133
 overview of firewall filters.....90
 for a VPLS routing instance
 overview of firewall filters.....90
 traffic forwarded or flooded to.....122
 for an aggregated Ethernet interface.....98
 for physical interfaces
 binding at the FPC level.....108
 binding at the PIC level.....110
 defining next-hop groups.....124
 displaying next-hop groups.....125
 order of precedence if bindings
 overlap.....96
 overview of named instances.....88
 overview of the global instance.....87
 global instance
 compared with other port mirroring
 sets.....94
 configuring.....101
 disabling.....112
 displaying.....111
 overview.....87
 mirroring duplicate packets to the same
 destination
 option to prevent101, 117
 multiple destinations
 defining next-hop-groups.....124
 displaying next-hop-groups.....125
 example.....140
 overview.....92
 multiple instances.....127

named instances.....	88
binding to a specific DPC or FPC.....	108
binding to ports grouped at the PIC level.....	110
compared with other port mirroring sets.....	94
defining.....	104
disabling.....	112
displaying.....	111
example with multiple instances.....	129
overview.....	88
<i>See also</i> firewall filters	
<i>See also</i> global instance	
overview.....	85
port-information-state-machine (tracing flag).....	289
port-migration-state-machine (tracing flag).....	289
port-mirroring, Layer 2 firewall filter application on PE routers.....	96
port-receive-state-machine (tracing flag) spanning-tree protocols.....	289
port-role-select-state-machine (tracing flag) spanning-tree protocols.....	289
port-role-transit-state-machine (tracing flag) spanning-tree protocols.....	289
port-state-transit-state-machine (tracing flag) spanning-tree protocols.....	289
port-transmit-state-machine (tracing flag) spanning-tree protocols.....	289
ppmd (tracing flag) spanning-tree protocols.....	289
priority statement spanning tree.....	281
priority-hold-time statement.....	282
usage guidelines.....	247
protection for spanning-tree protocols enhanced loop protection example.....	255
loop protection configuration guidelines..... configuring..... overview.....	234 235 233
protocol statement.....	283
protocols Layer 2 protocol tunneling configuration guidelines.....	252
protocols statement.....	284
ptopo-configuration-maximum-hold-time statement.....	41
ptopo-configuration-trap-interval statement.....	42
R	
Rapid Spanning-Tree Protocol See RSTP	
registration statement MVRP.....	79
revision-level statement.....	285
root protection spanning-tree protocols configuration guidelines..... configuring..... overview.....	236 237 236
routing instances complete configuration..... logical systems for configuration guidelines..... configuring..... example..... overview..... types used in Layer 2 networking instance-type layer2-control..... instance-type virtual-switch..... instance-type vpls..... types used in Layer 2 networking.....	15 27 27 28 13 18 17 16 14
routing instances for VPLS VLAN identifier list.....	152
routing-instances statement usage guidelines.....	15
routing-interface statement.....	185
RSTP BPDU overview..... configuring..... forcing to run as IEEE 802.1D STP configuration guidelines..... reverting from IEEE 802.1D STP..... nonstop bridging support..... overview.....	206 217 210 210 206 205
RSTP configuration BPDU destination MAC address..... bridge identifier for different routing instances..... bridge priority..... interface is edge port until BPDU is received..... interface participating in the instance..... interval for root bridge sending configuration BPDUs.....	211 212 212 215 213 212

- link cost for determining designated bridge and port.....214
 - link mode to identify point-to-point links.....215
 - maximum interval between arrival of hello BPDUs.....212
 - priority of interface to become root port.....214
 - root protect option
 - configuration guidelines.....236
 - configuring.....237
 - overview.....236
 - time bridge port remains in listening, learning state.....213
 - VLAN
 - link mode to identify point-to-point links.....215
 - rstp statement.....286
 - usage guidelines.....217
- S**
- snooping.....8, 23
 - socket (tracing flag)
 - MVRP.....80
 - Spanning-Tree Protocol (IEEE 802.1D) *See* STP
 - spanning-tree protocols
 - and multicast snooping
 - configuring.....23
 - example.....24
 - BPDUs overview.....206
 - BPDUs protection
 - configuration guidelines.....238, 240
 - overview.....238
 - in logical systems
 - configuration guidelines.....209
 - loop protection
 - configuration guidelines.....234
 - configuring.....235
 - example.....255
 - overview.....233
 - nonstop bridging support.....206
 - overview.....205
 - tracing protocol operations
 - configuration guidelines.....216
 - traceoptions statement.....288
 - VPLS root topology change actions
 - backup bridge priority.....245
 - bridge flush of MAC cache.....247
 - example.....256
 - global spanning tree behavior.....247
 - hold time for switching priority.....245
 - overview.....242
 - system identifier for bridges in the ring.....246
 - topology.....243
 - VLAN spanning tree behavior.....249
 - state-machine (tracing flag)
 - MVRP.....80
 - state-machine-variables (tracing flag)
 - spanning-tree protocols.....289
 - static-mac statement.....186
 - usage guidelines.....167
 - STP (IEEE 802.1D)
 - Layer 2 Protocol Tunneling
 - overview.....250
 - support, technical *See* technical support
 - switch-options statement.....187
 - syntax conventions.....xxv
 - system-id statement.....287
 - usage guidelines.....247
- T**
- technical support
 - contacting JTAC.....xxvi
 - timers (tracing flag)
 - MVRP.....80
 - spanning-tree protocols.....289
 - topology-change-state-machine (tracing flag)
 - spanning-tree protocols.....289
 - traceoptions statement
 - LLDP.....43
 - usage guidelines.....34
 - MVRP.....80
 - spanning-tree protocols.....288
 - usage guidelines.....216
 - tracing flags
 - all.....80, 288
 - all-failures
 - spanning-tree protocols.....288
 - bpdu.....288
 - bridge-detection-state-machine.....288
 - error
 - MVRP.....80
 - events
 - MVRP.....80
 - spanning-tree protocols.....288
 - pdu
 - MVRP.....80
 - port-information-state-machine.....289
 - port-migration-state-machine.....289

port-receive-state-machine		
spanning-tree protocols.....	289	
port-role-select-state-machine		
spanning-tree protocols.....	289	
port-role-transit-state-machine		
spanning-tree protocols.....	289	
port-state-transit-state-machine		
spanning-tree protocols.....	289	
port-transmit-state-machine		
spanning-tree protocols.....	289	
ppmd		
spanning-tree protocols.....	289	
socket		
MVRP.....	80	
state-machine		
MVRP.....	80	
state-machine-variables		
spanning-tree protocols.....	289	
timers		
MVRP.....	80	
spanning-tree protocols.....	289	
topology-change-state-machine		
spanning-tree protocols.....	289	
tracing operations		
LLDP.....	34, 43	
spanning-tree protocols		
configuration guidelines.....	216	
STP.....	288	
transmit-delay statement		
LLDP.....	45	
Trio		
Layer 2 parity.....	7	
tunnel interfaces		
configuring, MX Series routers.....	172	
tunnel-services statement.....	188	
usage guidelines.....	172	
V		
virtual switch		
configuring.....	148, 160	
dual VLAN tags.....	191	
routing interface.....	185	
VLAN identifier.....	189	
with VPLS ports.....	163	
virtual switch routing instance		
minimum configuration.....	17	
virtual switch routing instances		
logical systems for		
configuration guidelines.....	27	
configuring.....	27	
example.....	28	
virtual-switch routing instances.....	17	
virtual-switch statement		
usage guidelines.....	160	
VLAN identifiers		
configuring.....	152	
VLAN Spanning-Tree Protocol See VSTP		
vlan statement.....	291	
VLAN Trunk Protocol See VTP		
vlan-id statement.....	189	
vlan-id-list		
usage guidelines.....	151	
vlan-id-list statement.....	190	
vlan-tags statement.....	191	
VPLS.....	9	
VPLS multihomed Layer 2 ring		
overview.....	242	
VPLS root topology change actions		
backup bridge priority.....	245	
bridge flush of MAC cache.....	247	
hold time for switching priority.....	245	
system identifier for bridges in the		
ring.....	246	
VPLS ports in a virtual switch.....	163	
VPLS root protection		
and multicast snooping		
overview.....	21	
VPLS root topology change actions		
configuring global spanning tree behavior.....	247	
configuring VLAN spanning tree behavior.....	249	
VPLS routing instance		
minimum configuration.....	16	
vpls routing instances.....	16	
vpls-flush-on-topology-change statement.....	293	
usage guidelines.....	247	
VSTP		
BPDU overview.....	206	
configuring.....	226	
forcing to run as IEEE 802.1D STP		
configuration guidelines.....	210	
reverting from IEEE 802.1D STP.....	210	
overview.....	205	
VLAN.....	292	

VSTP configuration	
instance interface	
edge port.....	215
link cost for determining designated bridge and	
port.....	214
link mode to identify point-to-point links.....	215
priority of interface to become root port.....	214
root protect option	
configuration guidelines.....	236
configuring.....	237
overview.....	236
VLAN	
interface participating a VLAN	
instance.....	213
maximum interval between arrival of hello	
BPDUs.....	212
time bridge port remains in listening,	
learning state.....	213
VLAN	
bridge priority.....	212
edge port.....	215
interval for root bridge sending	
configuration BPDUs.....	212
link cost for determining designated bridge	
and port.....	214
priority of interface to become root	
port.....	214
VLAN root protect configuration	
guidelines.....	236
VLAN root protect configuring.....	237
VLAN root protect overview.....	236
vstp statement.....	294
usage guidelines.....	226
VTP	
Layer 2 Protocol Tunneling	
overview.....	250
Layer 2 protocol tunneling	
configuring.....	252

Index of Statements and Commands

A

advertisement-interval statement.....37

B

backup-bridge-priority statement.....260

bandwidth statement.....175

bpdu-block statement.....261

bpdu-block-on-edge statement.....261

bpdu-destination-mac-address statement

 MVRP.....71

 spanning tree.....262

bpdu-timeout-action statement.....263

bridge-domains statement.....176

bridge-options statement.....177

bridge-priority statement.....264

C

configuration-name statement.....265

cost statement.....266

D

disable statement

 LLDP.....38

 mstp.....267

disable-timeout statement.....267

domain-type statement.....178

E

edge statement.....268

extended-system-id statement.....269

F

force-version statement.....269

forward-delay statement.....270

G

global-mac-limit statement.....199

global-mac-statistics statement.....200

global-mac-table-aging-time statement.....200

global-no-mac-learning statement.....201

H

hello-time statement.....271

hold-multiplier statement.....38

I

interface statement

 BPDU blocking.....272

 bridge domain.....178

 Layer 2 protocol tunneling.....272

 LLDP.....39

 MVRP.....72

 spanning tree.....273

interface-mac-limit statement.....179

J

join-timer statement

 MVRP.....73

L

l2-learning statement.....201

layer2-control statement.....274

leave-timer statement

 MVRP.....75

leaveall-timer statement

 MVRP.....74

lldp statement

 LLDP.....40

lldp-configuration-notification-interval

 statement.....41

M

mac-rewrite statement.....275

mac-statistics statement.....180

mac-table-size statement.....181

max-age statement.....275

max-hops statement.....276

mode statement.....277

msti statement.....278

mstp statement.....279

mvrp statement.....76

N

no-dynamic-vlan statement	
MVRP.....	77
no-irb-layer-2-copy statement.....	182
no-mac-learning statement.....	183
no-root-port statement.....	280

P

packet-action statement.....	184
point-to-point statement	
MVRP.....	78
priority statement	
spanning tree.....	281
priority-hold-time statement.....	282
protocol statement.....	283
protocols statement.....	284
ptopo-configuration-maximum-hold-time	
statement.....	41
ptopo-configuration-trap-interval statement.....	42

R

registration statement	
MVRP.....	79
revision-level statement.....	285
routing-interface statement.....	185
rstp statement.....	286

S

static-mac statement.....	186
switch-options statement.....	187
system-id statement.....	287

T

traceoptions statement	
LLDP.....	43
MVRP.....	80
spanning-tree protocols.....	288
transmit-delay statement	
LLDP.....	45
tunnel-services statement.....	188

V

vlan statement.....	291
vlan-id statement.....	189
vlan-id-list statement.....	190
vlan-tags statement.....	191
vpls-flush-on-topology-change statement.....	293
vstp statement.....	294