JUNOSe™ Software
for E-series™ Routing Platforms

Multicast Routing Configuration Guide

Release 9.0.x

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

Part Number: 162-01846-00, Revision A00

JUNOSe™ Software for E-series™ Routing Platforms Multicast Routing Configuration Guide, Release 9.0.x
Writing: Mark Barnard, Diane Florio, Bruce Gillham, Sarah Lesway-Ball, Brian Wesley Simmons, Fran Singer
Editing: Ben Mann, Fran Mues
Illustration: Nathaniel Woodward
Cover Design: Edmonds Design

Revision History
29 February 2008—Revision 1

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

Software License

The terms and conditions for using this software are described in the software license contained in the acknowledgment to your purchase order or, to the extent applicable, to any reseller agreement or end-user purchase agreement executed between you and Juniper Networks. By using this software, you indicate that you understand and agree to be bound by those terms and conditions.

Generally speaking, the software license restricts the manner in which you are permitted to use the software and may contain prohibitions against certain uses. The software license may state conditions under which the license is automatically terminated. You should consult the license for further details.

For complete product documentation, please see the Juniper Networks Web site at www.juniper.net/techpubs.

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 Juniper Networks, Inc. and its subsidiaries (collectively "Juniper"), and 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, and updates and releases of such software, for which Customer has paid the applicable license or support fees to Juniper or an authorized Juniper reseller. "Embedded Software" means Software which Juniper has embedded in the Juniper 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 the 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 software on multiple computers requires multiple licenses, regardless of whether such computers 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 the Embedded Software on non-Juniper equipment; (j) use the 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 for the Software are exclusive of taxes, withholdings, duties, or levies (collectively "Taxes"). Customer shall be responsible for paying Taxes arising from the purchase of the license, or importation or use of the Software.

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 présentés confirment leur volonté que cette convention de même que tous les documents y compris tout avis qui s'y rattaché, soient redigés en langue anglaise. (Translation: The parties confirm that this Agreement and all related documentation is and will be in the English language)).

Table of Contents

About This Guide

This preface provides the following guidelines for using the *JUNOSe™ Software for E-series™ Routing Platforms Multicast Routing Configuration Guid*e:

Objectives

This guide provides the information you need to configure multicast routing for IP and IPv6 on your E-series router.

An E-series router is shipped with the latest system software installed. If you need to install a future release or reinstall the system software, refer to the procedures in *JUNOSe System Basics Configuration Guide, Chapter 3, Installing JUNOSe Software*.

NOTE: If the information in the latest *JUNOSe Release Notes* differs from the information in this guide, follow the *JUNOSe Release Notes*.

Audience

This guide is intended for experienced system and network specialists working with E-series routers in an Internet access environment.

E-series Routers

Seven models of E-series routers are available:

- E120 router
- E320 router
- ERX-1440 router
- ERX-1410 router
- ERX-710 router
- ERX-705 router
- ERX-310 router

All models use the same software. For information about all models except the E120 router and the E320 router, see *ERX Hardware Guide, Chapter 1, ERX Overview*. For information about the E120 router and the E320 router, see *E120 and E320 Hardware Guide*, *Chapter 1*, *E120 and E320 Overview*.

In the E-series documentation, the term ERX-14xx models refers to both the ERX-1440 router and the ERX-1410 router. Similarly, the term ERX-7xx models refers to both the ERX-710 router and the ERX-705 router. The terms ERX-1440 router, ERX-1410 router, ERX-710 router, ERX-705 router, ERX-310 router, E120 router, and E320 router refer to the specific models.

Documentation Conventions

Table 1 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.

Table 2 defines text conventions used in this guide and the syntax conventions used primarily in the *JUNOSe Command Reference Guide*. For more information about command syntax, see *JUNOSe System Basics Configuration Guide, Chapter 2, Command-Line Interface*.

Table 2: Text and Syntax Conventions

Convention	Description	Examples
Text Conventions		
Bold text like this	Represents commands and keywords in text.	■ Issue the **clock source** command. ■ Specify the keyword **exp-msg**.
Bold text like this	Represents text that the user must type.	host1(config)#**traffic class low-loss1**

`Fixed-width text like this`	Represents information as displayed on your terminal's screen.	`host1#show ip ospf 2`
		`Routing Process OSPF 2 with Router ID 5.5.0.250`
		`Router is an Area Border Router (ABR)`

Italic text like this	■ Emphasizes words. ■ Identifies variables. ■ Identifies chapter, appendix, and book names.	■ There are two levels of access, *user* and *privileged*. ■ *clusterId*, *ipAddress*. ■ *Appendix A*, *System Specifications*.
Plus sign (+) linking key names	Indicates that you must press two or more keys simultaneously.	Press Ctrl + b.
Syntax Conventions in the *Command Reference Guide*		
Plain text like this	Represents keywords.	terminal length
Italic text like this	Represents variables.	*mask*, *accessListName*
| (pipe symbol)	Represents a choice to select one keyword or variable to the left or right of this symbol. (The keyword or variable can be either optional or required.)	diagnostic | line
[ ] (brackets)	Represent optional keywords or variables.	[internal | external]
[ ]* (brackets and asterisk)	Represent optional keywords or variables that can be entered more than once.	[level1 | level2 | l1]*
{ } (braces)	Represent required keywords or variables.	{ permit | deny } { in | out } { *clusterId* | *ipAddress* }

Related E-series and JUNOSe Documentation

The E-series and JUNOSe documentation set consists of several hardware and software guides, which are available in electronic and printed formats.

E-series and JUNOSe Documents

Table 3 lists and describes the E-series and JUNOSe document set. For a complete list of abbreviations used in this document set, along with their spelled-out terms, see *JUNOSe System Basics Configuration Guide, Appendix A, Abbreviations and Acronyms*.

Table 3: Juniper Networks E-series and JUNOSe Technical Publications

Document	Description
E-series Hardware Documentation	
E120 and E320 Quick Start Guide	Shipped in the box with all new E120 and E320 routers. Provides the basic procedures to help you get the routers up and running quickly.
E120 and E320 Hardware Guide	Provides the necessary procedures for getting E120 routers and E320 routers operational, including information about: ■ Installing the chassis and modules ■ Connecting cables ■ Powering up the routers ■ Configuring the routers for management access ■ Troubleshooting common issues Describes switch route processor (SRP) modules, line modules, and I/O adapters (IOAs) available for E120 and E320 routers.
E120 and E320 Module Guide	Provides detailed specifications for line modules and IOAs in E120 and E320 routers, and information about the compatibility of these modules with JUNOSe software releases. Lists the layer 2 protocols, layer 3 protocols, and applications that line modules and their corresponding IOAs support. Provides module LED information.
E-series Installation Quick Start poster or *ERX Quick Start Guide*	Shipped in the box with all new ERX routers. Provides the basic procedures to help you get an ERX router up and running quickly.
ERX Hardware Guide	Provides the necessary procedures for getting ERX-14xx models, ERX-7xx models, and ERX-310 routers operational, including information about: ■ Installing the chassis and modules ■ Connecting cables ■ Powering up the routers ■ Configuring the routers for management access ■ Troubleshooting common issues Describes switch route processor (SRP) modules, line modules, and I/O modules available for the ERX routers.
ERX Module Guide	Provides detailed specifications for line modules and I/O modules in ERX-14xx models, ERX-7xx models, and ERX-310 routers, and information about the compatibility of these modules with JUNOSe software releases. Lists the layer 2 protocols, layer 3 protocols, and applications that line modules and their corresponding I/O modules support. Provides module LED information.

Table 3: Juniper Networks E-series and JUNOSe Technical Publications (continued)

Document	Description
ERX End-of-Life Module Guide	Provides an overview and description of ERX modules that are end-of-life (EOL) and can no longer be ordered for the following routers: ■ ERX-7xx models ■ ERX-14xx models ■ ERX-310 router
JUNOSe Software Guides	
JUNOSe System Basics Configuration Guide	Provides information about: ■ Planning and configuring your network ■ Using the command-line interface (CLI) ■ Installing JUNOSe software ■ Configuring the Simple Network Management Protocol (SNMP) ■ Managing the router and its modules, including the use of high availability (HA) for SRP redundancy ■ Configuring and running a unified in-service software upgrade (ISSU) ■ Configuring passwords and security ■ Configuring the router clock ■ Configuring virtual routers
JUNOSe Physical Layer Configuration Guide	Explains how to configure, test, and monitor physical layer interfaces.
JUNOSe Link Layer Configuration Guide	Explains how to configure and monitor static and dynamic link layer interfaces.
JUNOSe IP, IPv6, and IGP Configuration Guide	Explains how to configure and monitor IP, IPv6 and Neighbor Discovery, and interior gateway protocols (RIP, OSPF, and IS-IS).
JUNOSe IP Services Configuration Guide	Explains how to configure and monitor IP routing services. Topics include: ■ Routing policies ■ Firewalls ■ Network Address Translation (NAT) ■ J-Flow statistics ■ Bidirectional forwarding detection (BFD) ■ Internet Protocol Security (IPSec) ■ Access Node Control Protocol (ANCP), also known as Layer 2 Control (L2C) ■ Digital certificates ■ IP tunnels ■ Virtual Router Redundancy Protocol (VRRP) ■ Mobile IP home agent
JUNOSe Multicast Routing Configuration Guide	Explains how to configure and monitor IP multicast routing and IPv6 multicast routing. Topics include: ■ Internet Group Management Protocol (IGMP) ■ Protocol Independent Multicast (PIM) ■ Distance Vector Multicast Routing Protocol (DVMRP) ■ Multicast Listener Discovery (MLD)

Table 3: Juniper Networks E-series and JUNOSe Technical Publications (continued)

Document	Description
JUNOSe BGP and MPLS Configuration Guide	Explains how to configure and monitor: ■ Border Gateway Protocol (BGP) routing ■ Multiprotocol Label Switching (MPLS) and related applications ■ Layer 2 services over MPLS ■ Virtual private LAN service (VPLS) ■ Layer 2 virtual private networks (L2VPNs)
JUNOSe Policy Management Configuration Guide	Explains how to configure, manage, and monitor customized policy rules for packet classification, forwarding, filtering, and flow rates. Also describes the packet mirroring feature, which uses secure policies.
JUNOSe Quality of Service Configuration Guide	Explains how to configure quality of service (QoS) features to queue, schedule, and monitor traffic flow. These features include: ■ Traffic classes and traffic-class groups ■ Drop, queue, QoS, and scheduler profiles ■ QoS parameters ■ Statistics
JUNOSe Broadband Access Configuration Guide	Explains how to configure and monitor a remote access environment, which can include the following features: ■ Authentication, authorization, and accounting (AAA) ■ Dynamic Host Configuration Protocol (DHCP) ■ Remote Authentication Dial-In User Service (RADIUS) ■ Terminal Access Controller Access Control System (TACACS+) ■ Layer 2 Tunneling Protocol (L2TP) ■ Subscriber management
JUNOSe System Event Logging Reference Guide	Describes the JUNOSe system logging feature and describes how to use the CLI to monitor your system's log configuration and system events.
JUNOSe Command Reference Guide A to M; JUNOSe Command Reference Guide N to Z	Together constitute the *JUNOSe Command Reference Guide*. Contain important information about commands implemented in the system software. Use to look up: ■ Descriptions of commands and command parameters ■ Command syntax ■ A command's related mode ■ Starting with JUNOSe Release 7.1.0, a history of when a command, its keywords, and its variables were introduced or added Use with the JUNOSe configuration guides.
JUNOSe Comprehensive Index	Provides a complete index of the JUNOSe software documentation set.
JUNOSe Glossary	Provides definitions for terms used in JUNOSe technical documentation.
Release Notes	
JUNOSe Release Notes	Provide the latest information about features, changes, known problems, resolved problems, and system maximum values. If the information in the *Release Notes* differs from the information found in the documentation set, follow the *Release Notes*. Release notes are included on the corresponding software CD and are available on the Web.

JUNOSe Configuration Guides

JUNOSe software configuration guides use a bottom-up approach to describe the relationship of layers, protocols, and interfaces in the configuration process. For more information, see *Layered Approach* in *JUNOSe System Basics Configuration Guide, Chapter 1, Planning Your Network*.

The chapters in JUNOSe software configuration guides typically include the following topics:

- Conceptual and overview information
- Information you need to know or tasks you need to perform before you begin
- Platform-specific issues you need to take into consideration
- Applicable references, such as RFCs and IETF draft documents, about the protocols and features supported by the router
- Required and optional tasks, as step-by-step procedures
- Descriptions and examples of the commands you use
- Illustrations of network topologies
- Examples of command sequences for configuration, testing, and monitoring activities
- Sample displays that result when you issue the **show** command

Obtaining Documentation

To obtain the most current version of all Juniper Networks technical documentation, see the products documentation page on the Juniper Networks Web site at http://www.juniper.net/.

To order printed copies of this manual and other Juniper Networks technical documents or to order a documentation CD, which contains this manual, contact your sales representative.

Copies of the Management Information Bases (MIBs) available in a software release are included on the software CDs and at http://www.juniper.net/.

Documentation Feedback

We encourage you to provide feedback, comments, and suggestions so that we can improve the documentation to better meet your needs. Send your comments to techpubs-comments@juniper.net, or fill out the documentation feedback form at http://www.juniper.net/techpubs/docbug/docbugreport.html. If you are using e-mail, be sure to include the following information with your comments:

- Document name
- Document part number
- Page number
- Software release version

Requesting Technical Support

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

- **JTAC Policies**—For a complete understanding of our JTAC procedures and policies, review the *JTAC User Guide* located at http://www.juniper.net/customers/support/downloads/710059.pdf
- **Product Warranties**—For product warranty information, visit http://www.juniper.net/support/warranty/
- **JTAC Hours of Operation**—The JTAC centers have resources available 24 hours a day, 7 days a week, 365 days a year.

Self-Help Online Tools and Resources

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

- Find CSC offerings: http://www.juniper.net/customers/support/
- Search for known bugs: http://www2.juniper.net/kb/
- Find product documentation: http://www.juniper.net/techpubs/
- Find solutions and answer questions using our Knowledge Base: http://kb.juniper.net/
- Download the latest versions of software and review release notes: http://www.juniper.net/customers/csc/software/

- Search technical bulletins for relevant hardware and software notifications: 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 Manager: http://www.juniper.net/cm/

To verify service entitlement by product serial number, use our Serial Number Entitlement (SNE) Tool located at 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 Manager 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 http://www.juniper.net/support/requesting-support.html

Part 1

Internet Protocol Version 4

Chapter 1

Configuring IPv4 Multicast

IPv4 multicast enables a device to send packets to a group of hosts rather than to a list of individual hosts. This chapter describes how to configure IP multicast on the E-series router; it contains the following sections:

Overview

IPv4 defines three types of addresses: *unicast*, *broadcast*, and *multicast*. Each type of address enables a device to send datagrams to selected recipients:

- A unicast address enables a device to send a datagram to a single recipient.
- A broadcast address enables a device to send a datagram to all hosts on a subnetwork.
- A multicast address enables a device to send a datagram to a specified set of hosts, known as a multicast group, in different subnetworks.

Multicast IP packets contain a class D address in the Destination Address fields of their headers. A class D address is the IP address of a multicast group. See *Chapter 2, Configuring IGMP* and *JUNOSe IP, IPv6, and IGP Configuration Guide, Chapter 1, Configuring IP*, for information about class D addresses.

IP multicast improves network efficiency by enabling a host to transmit a datagram to a targeted group of receivers. For example, for a host to send a large video clip to a group of selected recipients would be time-consuming to unicast the datagram to each recipient individually. If the host broadcasts the video clip throughout the network, network resources are not available for other tasks. The host uses only the resources it needs when multicasting the datagram.

Routers use multicast routing algorithms to determine the best route and transmit multicast datagrams throughout the network. E-series routers support a number of IP multicast protocols on virtual routers (VRs). Each VR handles the interoperability of IP multicast protocols automatically. To start multicast operation on a VR, you access the context for that VR and configure the desired protocols on the selected interfaces. Table 4 describes the function of each protocol that the router supports.

Table 4: Function of Multicast Protocols on a Router

Protocol	Function
Internet Group Membership Protocol (IGMP)	Discovers hosts that belong to multicast group.
Protocol Independent Multicast Protocol (PIM)	Discovers other multicast routers to receive multicast packets.
Distance Vector Multicast Routing Protocol (DVMRP)	Routes multicast datagrams within autonomous systems.
BGP Multicasting Protocol	Routes multicast datagrams between autonomous systems.

The router supports up to 16,384 multicast forwarding entries (multicast routes) at any time.

Reverse-Path Forwarding

IP multicasting uses reverse path forwarding (RPF) to verify that a router receives a multicast packet on the correct incoming interface. The RPF algorithm enables a router to accept a multicast datagram only on the interface from which the router sends a unicast datagram to the source of the multicast datagram.

When the router receives a multicast datagram from a source for a group, the router verifies that the packet was received on the correct RPF interface. If the packet was not received on the correct interface, the router discards the packet. Only packets received on the correct RPF interface are considered for forwarding to downstream receivers.

When operating in sparse-mode, the routers perform an RPF lookup to identify the upstream router from which to request the data and then send join messages for the multicast stream only to that router.

When operating in dense-mode, routers that have multiple paths to the source of the multicast stream initially receive the same stream on more than one interface. In this case, the routers perform an RPF lookup to identify multicast data streams that are not arriving on the best path and send prune messages to terminate these flows.

The RPF lookup need not always be towards the source of the multicast stream. The lookup is done towards the source only when the router is using a source-rooted tree to receive the multicast stream. If the router uses a shared tree instead, the RPF lookup is toward a rendezvous point and not toward the source of the multicast stream.

Multicast Packet Forwarding

Multicast packet forwarding is based on the source (S) of the multicast packet and the destination multicast group address (G). For each (S,G) pair, the router accepts multicast packets on an incoming interface (IIF), which satisfies the RPF check (RPF-IIF). The router drops packets received on IIFs other than the RPF-IIF and notifies the routing protocols that a packet was received on the wrong interface.

The router forwards packets received on the RPF-IIF to a list of outgoing interfaces (OIFs). The list of OIFs is determined by the exchange of routing information and local group membership information. The router maintains mappings of (S,G, IIF) to {OIF1, OIF2…} in the multicast routing table.

You can enable two or more multicast protocols on an IIF. However, only one protocol can forward packets on that IIF. The protocol that forwards packets on an IIF *owns* that IIF. A multicast protocol that owns an IIF also owns the (S,G) entry in the multicast routing table.

Platform Considerations

For information about modules that support IP multicasting on the ERX-7xx models, ERX-14xx models, and the ERX-310 router:

- See *ERX Module Guide, Table 1, Module Combinations* for detailed module specifications.
- See *ERX Module Guide, Appendix A, Module Protocol Support* for information about the modules that support IP multicasting.

For information about modules that support IP multicasting on the E120 router and the E320 router:

- See *E120 and E320 Module Guide, Table 1, Modules and IOAs* for detailed module specifications.
- See *E120 and E320 Module Guide, Appendix A, IOA Protocol Support* for information about the modules that support IP multicasting.

References

For more information about IP multicast, see the following resources:

- A "traceroute" Facility for IP Multicast—draft-ietf-idmr-traceroute-ipm-07.txt (January 2001 expiration)
- RFC 2858—Multiprotocol Extensions for BGP-4 (June 2000)
- RFC 2932—IPv4 Multicast Routing MIB (October 2000)
- RFC 3292—General Switch Management Protocol (GSMP) V3 (June 2002)

NOTE: IETF drafts are valid for only 6 months from the date of issuance. They must be considered as works in progress. Refer to the IETF Web site at http://www.ietf.org for the latest drafts.

Before You Begin

You can configure multicasting on IPv4 and IPv6 interfaces.

For information about configuring IP interfaces, see *JUNOSe IP, IPv6, and IGP Configuration Guide, Chapter 1, Configuring IP*. For information about configuring IPv6 interfaces, see *JUNOSe IP, IPv6, and IGP Configuration Guide, Chapter 2, Configuring IPv6*.

For information about configuring multicast on IPv6 interfaces, see *Chapter 5, Configuring IPv6 Multicast*.

Configuring the Switch Fabric Bandwidth

By default, the switch fabric for the ERX-1440, ERX-310, E120, and E320 routers uses a bandwidth weighting ratio of 15:2 for multicast-to-unicast weighted round robin (WRR). In the absence of strict-priority traffic, and when both unicast and multicast traffic compete for switch fabric bandwidth, the switch fabric allocates 15/17ths of the available bandwidth to multicast traffic and 2/17ths of the available bandwidth to unicast traffic.

You can use the **fabric weights** command to change the ratio for multicast-to-unicast traffic on the router switch fabric. For more information about the **fabric weights** command, see *JUNOSe System Basics Configuration Guide, Chapter 5, Managing the System*.

Enabling IP Multicast

In this implementation, IP multicast works on virtual routers (VRs). By default, IP multicast is disabled on a VR. To enable IP multicast on a VR, access the context for a VR, and then issue the **ip multicast-routing** command.

ip multicast-routing

- Use to enable IP multicast routing on the VR.
- By default, IP multicast is disabled on the VR. In the disabled state, all multicast protocols are disabled, and the VR forwards no multicast packets.
- Example

  host1(config)#**ip multicast-routing**

- Use the **no** version to disable IP multicast routing on the VR (the default).

Defining Static Routes for Reverse-Path Forwarding

Use the **ip rpf-route** command to define reverse-path forwarding (RPF) to verify that a router receives a multicast packet on the correct incoming interface.

ip rpf-route

- Use to customize static routes that the router may use for RPF.
- Specify the IP address and subnet mask of the destination network.
- Specify either a next-hop IP address or an interface type and specifier, such as atm 3/0. For details about interface types and specifiers, see *JUNOSe Command Reference Guide, About This Guide*.
- Optionally, specify the distance (number of hops) to the next-hop address.
- Optionally, specify a route's tag number to identify a particular route in the routing table.
- Example

  host1(config)#**ip rpf-route 11.1.0.0 255.255.0.0 atm4/1.1 56 tag 25093**

- Use the **no** version to remove the static route.

Displaying Available Routes for Reverse-Path Forwarding

Use the **show ip rpf-route** command to display all available routes, only the routes to a particular destination, or routes associated with a specific unicast protocol that the router can use for Reverse-Path Forwarding (RPF).

show ip rpf-route

- Use to display routes that the router can use for RPF.
- Specify the IP address and the network mask to view routes to a particular destination.
- Specify a unicast routing protocol to view routes associated with that protocol.
- Field descriptions
  - Prefix—Value of the logical AND of the IP address of the destination network and the subnet address
  - Length—Length of the subnet mask in bits
  - Type—Protocol type for the interface
    - Connect—Subnet directly connected to the interface
    - Static—Static route
    - *protocol-name*—Route learned through the named protocol
  - Next Hop—IP address of the next hop for this route
  - Dist—Distance configured for this route
  - Met—Learned or configured cost associated with this route
  - Intf—Type of interface and interface specifier for the next hop. For details about interface types and specifiers, see *JUNOSe Command Reference Guide, About This Guide*.
- Example 1

```
host1#show ip rpf-route
Protocol/Route type codes:
  I1- ISIS level 1, I2- ISIS level2,
  I- route type intra, IA- route type inter, E- route type external,
  i- metric type internal, e- metric type external,
  O- OSPF, E1- external type 1, E2- external type2,
  N1- NSSA external type1, N2- NSSA external type2
  L- MPLS label, V- VR/VRF, *- indirect next-hop

  Prefix/Length    Type    Next Hop       Dist/Met   Intf
  -------------    ----    --------       --------   ------
  10.10.0.112/32    Static  192.168.1.1     1/1        fastEthernet0/0
  10.1.1.0/24       Connect 10.1.1.1        0/1        atm3/0.100
  25.25.25.25/32    Connect 25.25.25.25     0/1        loopback0
```

- Example 2

```
host1#show ip rpf-route static
Protocol/Route type codes:
  I1- ISIS level 1, I2- ISIS level2,
  I- route type intra, IA- route type inter, E- route type external,
  i- metric type internal, e- metric type external,
  O- OSPF, E1- external type 1, E2- external type2,
  N1- NSSA external type1, N2- NSSA external type2
  L- MPLS label, V- VR/VRF, *- indirect next-hop

 Prefix/Length  Type   Next Hop   Dist/Met  Intf
 -------------  ----   --------   --------  --------------
10.10.0.112/32  Static 192.168.1.1  1/1     fastEthernet0/0
```

Enabling and Disabling RPF Checks

By default, the router accepts multicast packets for each Source, Group (S,G) pair on an incoming interface (IIF), which satisfies the RPF check (RPF-IIF). When the router performs RPF checks, only the interface that first accepts traffic for an (S,G) pair accepts subsequent traffic for that pair. If traffic stops arriving on that interface and starts arriving on another interface, the router does not accept or forward the traffic.

Some network configurations require the router to accept traffic on any interface. To do so, you can disable the RPF check on a specified set of (S,G) pairs by issuing the **ip multicast-routing disable-rpf-check** command.

When you disable RPF checks, the router accepts multicast packets for (S,G) pairs on any incoming interface. When the router has added the new route to its multicast routing table, it then accepts multicast packets for these pairs on any interface in the virtual router and forwards them accordingly. Multicast routes established before you issue this command are not affected.

ip multicast-routing disable-rpf-check

- Use to disable RPF checks for specified (S,G) pairs.
- Specify a standard IP access list that defines the (S,G) pairs.
- Example

  host1(config)#**ip multicast-routing disable-rpf-check boston-list**

- Use the **no** version to restore the default, in which the router performs RPF checks for all (S,G) pairs.

Using Unicast Routes for RPF

You can specify that IS-IS, OSPF, or RIP routes be available for RPF. Routes available for RPF appear in the multicast view of the routing table.

ip route-type

- Use to specify whether IS-IS, OSPF, or RIP routes are available only for unicast forwarding, only for multicast RPF checks, or for both.
- Use the **show ip rpf-routes** command to view the routes available for RPF.
- By default, IS-IS, OSPF, and RIP routes are available both for unicast forwarding and multicast reverse-path forwarding checks.
- Example

  host1(config)#**router ospf**
  host1(config-router)#**ip route-type multicast**

- There is no **no** version.

Defining Permanent IP Multicast Forwarding Entries

An mroute is a multicast traffic flow (a (Source, Group) entry used for forwarding multicast traffic). By default, forwarding mroutes (with a valid RPF incoming interface) are timed out if data for them is not received for 210 seconds. However, you can specify an mroute as permanent by using the **ip multicast-routing permanent-mroute** command.

ip multicast-routing permanent-mroute

- Use to specify that any newly created mroutes that match the specified access-list do not time out.
- Using this command does not change existing mroutes.
- Permanent mroutes are removed if a topology change occurs that affects the mroute.
- Permanent mroutes may be removed due to certain protocol actions (for example, PIM sparse-mode switching from shared to shortest-path tree).
- Outgoing interface lists of permanent mroutes may change due to protocol actions.
- Example

  host1(config)#**ip multicast-routing permanent-mroute routes1**

- Use the **no** version to prevent any new mroutes from becoming permanent. To remove existing permanent mroutes, use the **clear ip mroute** command.

Defining a Multicast Bandwidth Map

Multicast interface-level admission control, port-level admission control, and QoS adjustment all use a single multicast bandwidth map. The multicast bandwidth map is a route map that uses the **set admission-bandwidth**, **set qos-bandwidth**, **set admission-bandwidth adaptive**, or **set qos-bandwidth adaptive** commands. The **adaptive** commands configure an autosense mechanism for measuring the multicast bandwidth.

NOTE: Even though you can include any of the preceding commands several times in a route map entry, only the last **admission-bandwidth** command or **qos-bandwidth** command in the bandwidth map is used. In other words, if you included the **set qos-bandwidth** command first and then the **set qos-bandwidth adaptive** command, the bandwidth map uses the **set qos-bandwidth adaptive** command.

Interface-level and port-level admission control is performed when an OIF on the interface or port is added to the mroute for a given (S,G) multicast data stream and the multicast bandwidth map contains a **set admission-bandwidth** or **set admission-bandwidth adaptive** action for that (S,G).

QoS adjustment is performed on the joining interface when an OIF is added to the mroute for a given (S,G) data stream and the multicast bandwidth map contains a **set qos-bandwidth** or **set qos-bandwidth adaptive** action for that (S,G).

You can prioritize the traffic by configuring a priority value for the <S, G> data stream on a physical port by issuing the **set priority** command. Dynamic multicast admission control enables only prioritized groups to join the interface after the configured priority limit is reached on the physical port. The system records the priority when a new <S, G> entry is created. For more information, see *Enabling Port Admission Bandwidth Control* on page 29.

NOTE: You can create a single route map with the **set admission-bandwidth** command, the **set qos-bandwidth** command, or both. However, creating an entry with only one of these **set** commands enables only that specific function for the matched address (that is, only multicast traffic admission control or only QoS adjustment). The same is true for the **adaptive** commands.

Using the Autosense Mechanism

Video bandwidth is typically considered to be a constant rate—2 Mbps for standard definition television (SDTV) and 10 Mbps for high definition television (HDTV). However, in reality, and depending on achievable video compression, the bit rate can vary. For example, HDTV streams (using MPEG4 or WM9 encoding) can vary between 6 Mbps (for low-action programs) to 10 Mbps (for a fast-paced, high-action programs). The autosense mechanism causes the bandwidth value, used for admission control and QoS adjustment, to be the actual measured rate of the stream. Using this feature to measure the actual bandwidth avoids the need to configure arbitrary bandwidth limits and enables a channel to be reassigned to a different (S, G) without requiring a bandwidth map to be changed.

How Adaptive Mode Works

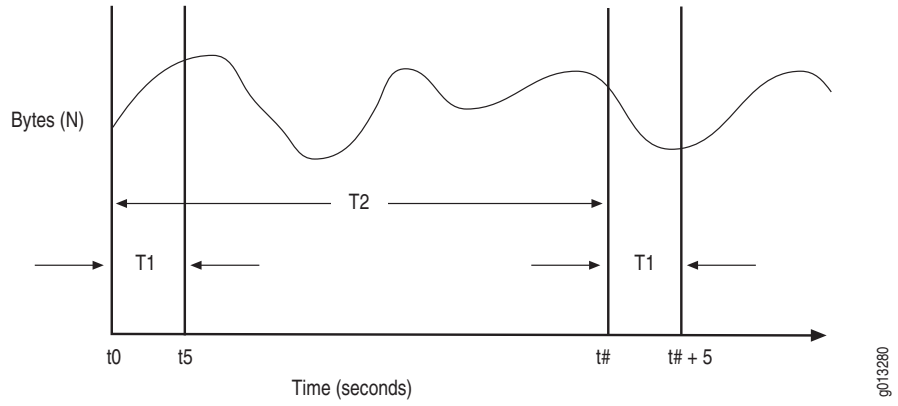
You configure the auto-sense mechanism in the multicast bandwidth using the **set admission-bandwidth adaptive** command, **set qos-bandwidth adaptive** command, or both. For example:

```
host1(config)#route-map mcast-bandwidths permit 10
host1(config-route-map)#match ip address sdtv
host1(config-route-map)#set admission-bandwidth adaptive
host1(config-route-map)#set qos-bandwidth adaptive
host1(config-route-map)#end
```

In this example, any stream with an (S,G) that matches the sdtv access list performs adaptive bandwidth detection for admission control and QoS adjustment.

A rate measurement mechanism runs on the ingress line card that polls the forwarding controller (FC) to obtain statistics for each mroute. This mechanism then reports the rate measurement to the SRP to update the bandwidth map. By computing the average bandwidth over a relatively short sampling period (T1; 5 seconds), the measurement approximates the peak bandwidth of the multicast stream.

As an example, assume that a new mroute (S1, G1) is added to the interface controller (IC) at time t0.



To calculate the measured bandwidth of a stream, the router uses the following equation:

$R = (N_{t+5} - N_t) / 5$

Where

R = Calculated bandwidth of the stream during each sampling interval

N_t = Bytes measured at the start of each sampling period (t seconds)

N_{t+5} = Bytes measured at the end of each sampling period (t + 5 seconds)

NOTE: When the mroute is first installed in the FC (at t = 0), R_0 is undetermined. For multicast admission control no joins are admitted until the first bandwidth measurement is computed (that is, for admission control, R_0 is considered to be infinite). Similarly, no QoS adjustment occurs until the first bandwidth measurement is computed (that is, for QoS adjustment, R_0 is considered to be zero [0]).

Using the previous graph as a reference, the first bandwidth rate (R_1) is determined by calculating the number of bytes received during the first sampling period, T_1. Mroute statistics are read at time t_0 (N_0) and at time t_5 (N_5) and the bytes received values are subtracted and divided by the sampling period T_1 to yield the average rate. This process is repeated every sampling interval, T_2, to yield rates R1, R2, R3, and so on.

The first two sampling interval calculations are as follows:

$R_1 = (N_5 - N_0)/5$

$R_2 = (N_{\#+5} - N_{\#})/5$

The router maintains a history of bandwidth measurements (H) for each mroute, up to a maximum of M measurements. The actual rate, R, reported to the SRP is the maximum rate measured in those H samples.

To minimize the IC to SRP traffic generated by the rate measurements, the IC reports a bandwidth change only when a newly computed rate (R#) differs from the current rate by a specified threshold. When R_s is computed at time t = 5 seconds, R is set to R_1. A rate update occurs whenever a newly calculated rate (R) differs from R_1 by at least a threshold value (specified as a percentage, P) of the measured peak bandwidth. This calculation is as follows:

R = Rt, if and only if the absolute value of (R - Rt) > P * R.

Table 5 lists values assigned to variables associated with this algorithm.

Table 5: Adaptive Mode Algorithm Values

Variable	Value	Units	Description
T1	5	Seconds	Sampling period; the time in which a sample is taken
T2	0	Seconds	Sampling interval; zero (0) seconds indicates continuous sampling
H	12	Samples	Number of history samples over which to compute measurement
M	12	Samples	Maximum number of samples maintained in history
P	1	Percent	Threshold value; percent difference by which a newly calculated rate must differ from the measured peak bandwidth before a rate update occurs

Multicast Bandwidth Map Example

The following example creates a multicast bandwidth map for both multicast traffic admission control and QoS adjustment:

NOTE: In this example, you can replace the **set admission-bandwidth** command and **set qos-bandwidth** command with their **adaptive** command counterparts.

1. Define a route-map using the **set admission-bandwidth** and **set qos-bandwidth** commands. You can optionally issue the **set priority** command.

   host1(config)#**route-map mcast-bandwidths permit 10**
   host1(config-route-map)#**match ip address sdtv**
   host1(config-route-map)#**set admission-bandwidth 2000000**
   host1(config-route-map)#**set qos-bandwidth 2000000**
   host1(config-route-map)#**set priority 100**
   host1(config-route-map)#**route-map mcast-bandwidths permit 20**
   host1(config-route-map)#**match ip address hdtv**
   host1(config-route-map)#**set admission-bandwidth 10000000**
   host1(config-route-map)#**set qos-bandwidth 10000000**
   host1(config-route-map)#**set priority 200**
   host1(config-route-map)#**end**

2. Define the access list for use by the **match ip address** command to match (S,G) and (*,G) entries.

   host1(config)#**access-list sdtv permit ip host 31.0.0.1 232.0.0.0 0.0.0.255**
   host1(config)#**access-list hdtv permit ip host 32.0.0.1 232.0.0.0 0.0.0.255**
   host1(config)#**access-list hdtv permit ip host 32.0.0.2 232.0.0.0 0.0.0.255**
   host1(config-route-map)#**end**

NOTE: You can also define a prefix-list or a prefix-tree for use by the **match ip address** command to match (S,G) and (*,G) entries.

For additional information about configuring QoS adjustment, see *Configuring Multicast QoS Adjustment* on page 15.

For additional information about configuring interface-level and port-level admission control, see *Blocking and Limiting Multicast Traffic* on page 26.

For additional information about creating route maps, see *JUNOSe IP Services Configuration Guide, Chapter 1, Configuring Routing Policy*.

set admission-bandwidth

- Use to set a multicast bandwidth for admission control.
- Use the **adaptive** keyword to define the bandwidth as adaptive (automatically sensed).
- Example

  host1(config-route-map)#**set admission-bandwidth 2000000**

- Use the **no** version to remove the set clause from a route map.

set priority

- Use to configure a priority value for the <S, G> data stream on a physical port.
- Dynamic multicast admission control enables only prioritized groups to join the interface after the configured priority limit is reached on the physical port. The system records the priority when a new <S, G> entry is created.
- Example

  host1(config-route-map)#**set priority 100**

- Use the **no** version to remove the priority value.

set qos-bandwidth

- Use to set a multicast bandwidth for QoS adjustment.
- Use the **adaptive** keyword to define the bandwidth as adaptive (automatically sensed).
- Example

  host1(config-route-map)#**set qos-bandwidth 10000000**

- Use the **no** version to remove the set clause from a route map.

Configuring Multicast QoS Adjustment

When the router uses multicast OIF mapping, any multicast streams that a subscriber receives bypass any configured QoS treatment for that subscriber interface. The Multicast QoS adjust feature provides a way in which the router can account for this multicast traffic.

NOTE: For additional information about how to configure OIF mapping, see *Configuring Group Outgoing Interface Mapping* on page 53.

The following sections provide two possible configuration cases for using multicast QoS adjustment.

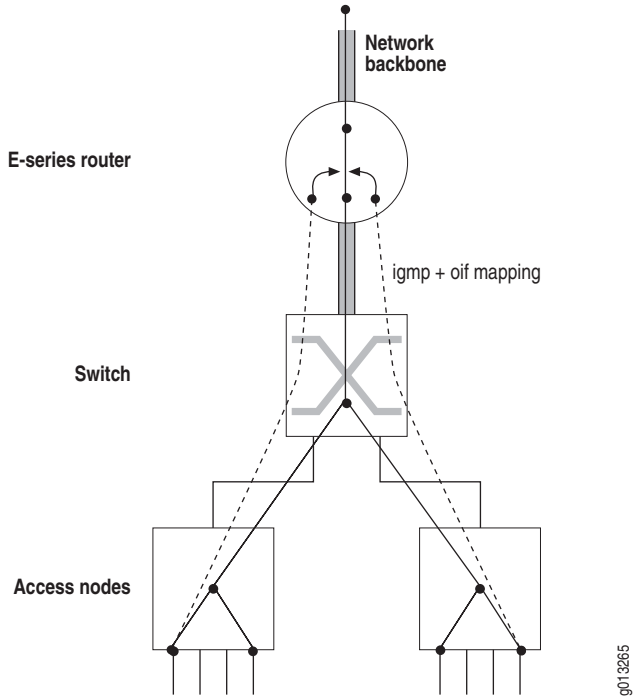
NOTE: For additional information about QoS adjustment, see *JUNOSe Quality of Service Configuration Guide, Chapter 26, Configuring IP Multicast Bandwidth Adjustment with QoS Parameters*.

Multicast OIF Mapping Case

Multicast OIF mapping enables the router to decrease the inefficiencies associated with replicating streams of multicast traffic. Using OIF maps, IGMP joins that the router receives on a subscriber interface can be mapped to a special interface for forwarding. This special interface can be on a different physical port or line module from that of the join interface.

Using this mapping function, the router can send a single copy of each multicast stream over the special interface and the access nodes are configured to perform any final replication to the subscribers and merge unicast and multicast data flows onto the subscriber interfaces as necessary. See Figure 1.

Figure 1: Multicast OIF Mapping



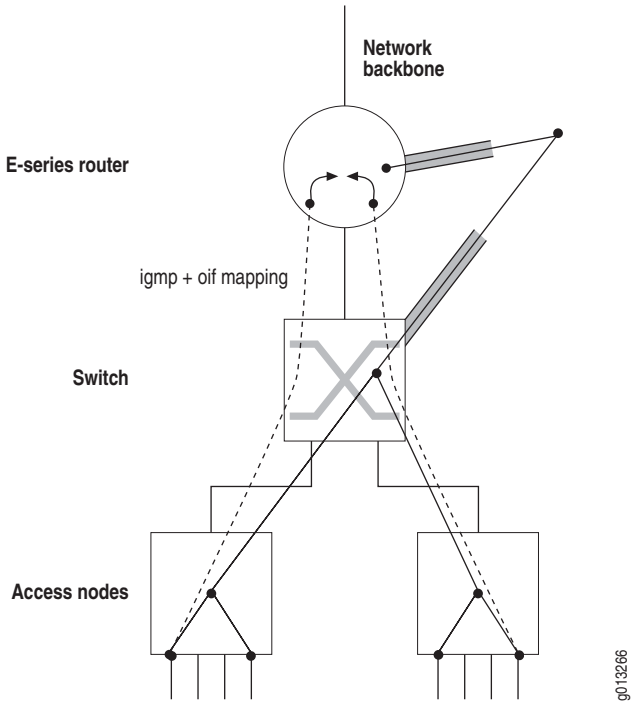
One disadvantage to using multicast OIF mapping is that the multicast traffic bypasses any QoS treatment that is applied to subscriber interfaces. Configuring QoS adjustment resolves this problem. (See *JUNOSe Quality of Service Configuration Guide, Chapter 24, Configuring a QoS Parameter* for additional information about configuring QoS adjustment.) With QoS adjustment configured, when a subscriber requests to receive a multicast stream (or, more appropriately, when an OIF is added to the mroute), the router reduces the unicast QoS bandwidth applied to the subscriber interface (that is, the join interface) by the amount of bandwidth for that multicast stream.

Multicast Traffic Receipt Without Forwarding

In this case, the router is not given the responsibility of forwarding multicast streams. Instead, the service provider arranges for the router to receive the multicast streams so the router can detect the flow and perform QoS adjustment. An OIF map is installed that maps the traffic streams to a loopback interface configured for IGMP version passive. This means that when the traffic is received, a null mroute is installed (that is, an mroute with an empty OIF list) and the router applies the QoS adjustment to the join interface. See Figure 2.

NOTE: Ensure that PIM-SM (or any other upstream multicast protocol) is informed of the group (or source-group) interest.

Figure 2: Multicast Traffic Receipt Without Forwarding



Activating Multicast QoS Adjustment Functions

The **ip multicast-routing bandwidth-map** command activates the specified bandwidth map. By activating the bandwidth map, this command also activates the multicast QoS adjustment function contained in the bandwidth map.

CAUTION: To activate multicast QoS adjustment, you must first create a bandwidth map. See *Defining a Multicast Bandwidth Map* on page 11 for details.

ip multicast-routing bandwidth-map

- Use to activate the QoS adjust function on the router.
- Example

  host1(config)#**ip multicast-routing bandwidth-map mcast-bandwidths**

- Use the **no** version to disable the multicast QoS adjustment function on the router.

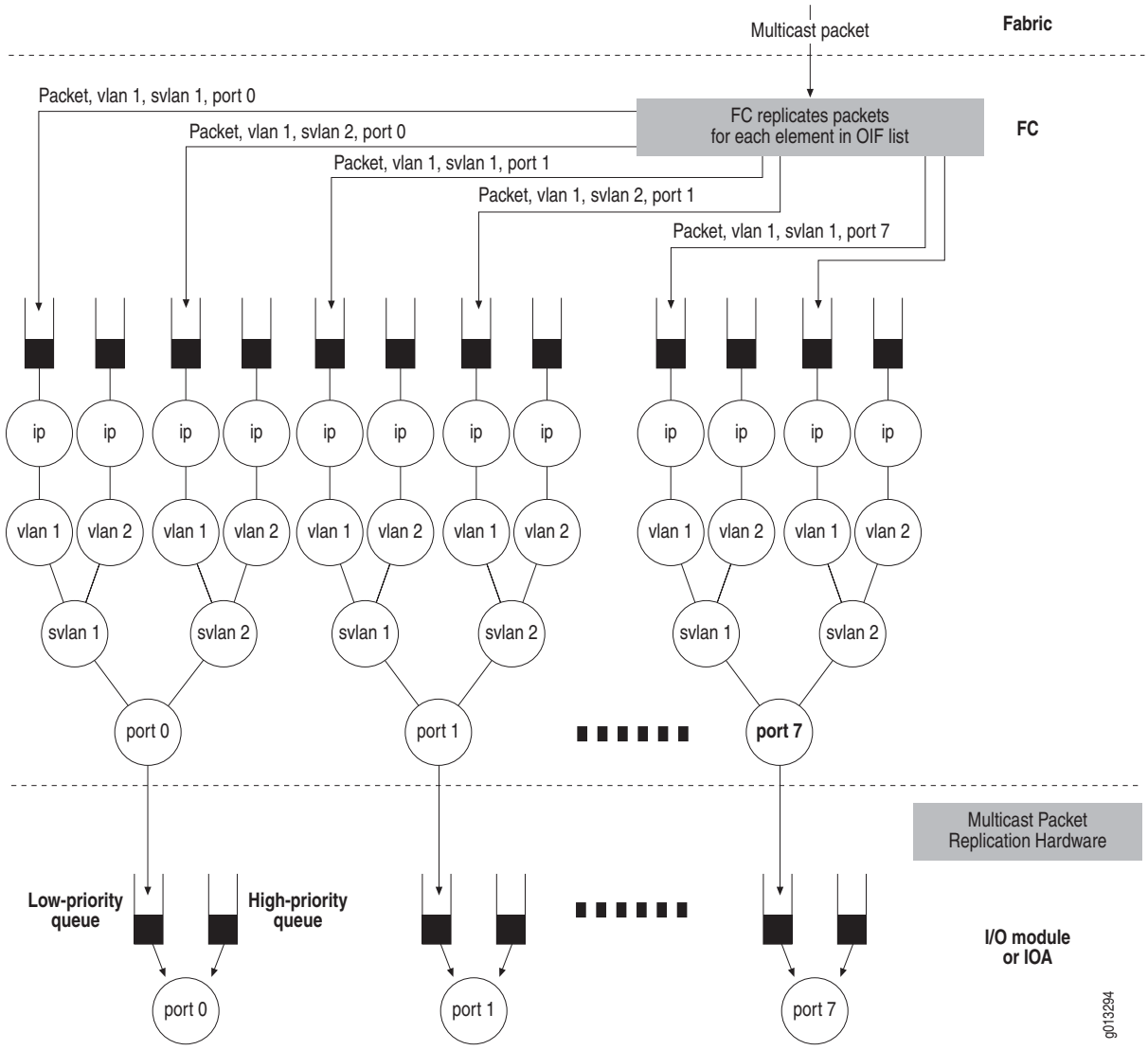
Configuring Hardware Multicast Packet Replication

You can configure IPv4 multicast to replicate packets to optimized hardware on a logical port instead of using the forwarding controller (FC) on the router.

The bandwidth between the line module and the I/O module or IOA on the E-series router is limited. A high-density Ethernet module provides eight physical ports that can consume the bandwidth between the line module and the I/O module or IOA before providing enough traffic to support egress line rate for all of these ports.

Figure 3 on page 18 displays how multicast traffic is typically replicated on the line module. Each of these replicated packets is transmitted from the line module to the I/O module or IOA.

Figure 3: Packet Flow Without Hardware Multicast Packet Replication



The hardware multicast packet replication feature enables you to configure multicast traffic for a VLAN or S-VLAN to be replicated on the I/O module or IOA so that only one copy of the packet is transmitted from the line module to the I/O module or IOA. Replication for each of the ports is performed on the I/O module or IOA.

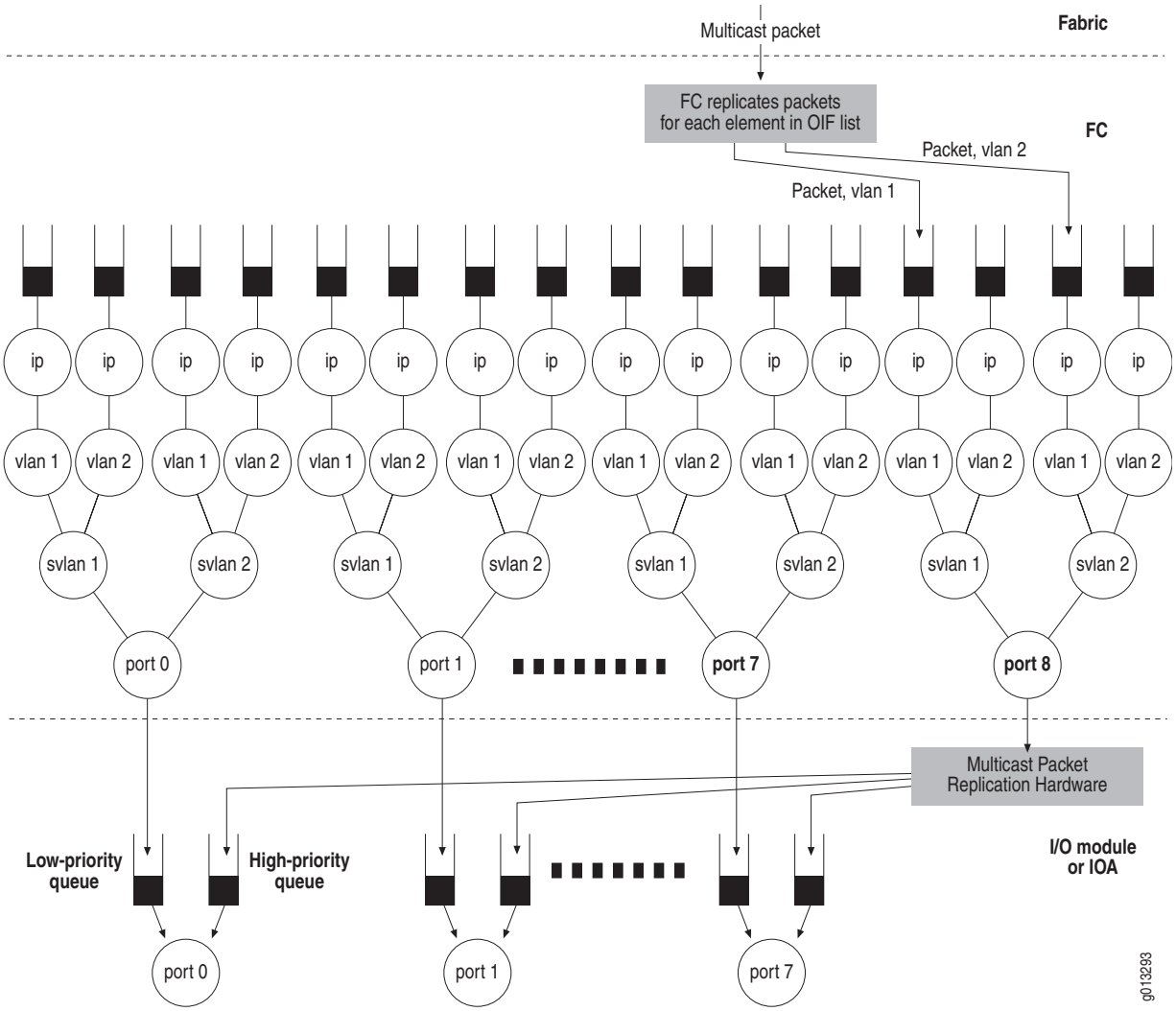
Configuring hardware multicast packet replication for high-density Ethernet is useful when you want to provide the same multicast stream out of some or all of the ports, such as for IP television (IPTV). Configuring hardware multicast packet replication enables you to:

- Reduce the number of packets sent from the FC to the module.
- Reduce the CPU consumed by the FC processing each elaboration of the packet.

You can use the additional bandwidth to increase the bandwidth of multicast traffic out of each of the Gigabit Ethernet ports.

Figure 4 displays the flow of a multicast packet using the hardware multicast packet feature.

Figure 4: Packet Flow with Hardware Multicast Packet Replication



Each high-density Ethernet module has eight physical ports, numbered 0–7. A logical port is available for the hardware multicast packet replication feature, numbered port 8.

JUNOSe tracks the OIFs in an mroute that have been redirected to use the hardware multicast packet replication hardware. The system accepts only egress multicast traffic to traverse the interface stack on the enabled port. The system drops unicast traffic that is routed to this port.

Each port on the I/O module or IOA displayed in Figure 4 has two queues. These queues are further down the egress path than the queues found on the line module and populated by the FC.

The low-priority queue is dedicated to packets that are received from the line module queues that are dedicated to the physical ports. This queue blocks when full and provides backpressure to the line module. This queue services unicast and multicast traffic that is not using the hardware multicast packet replication feature.

The high-priority queue is dedicated to packets that are received from the line module queue for port 8. This queue is serviced at a higher priority than the first queue, and drops packets when full.

For more information about high-density Ethernet, see *JUNOSe Physical Layer Configuration Guide, Chapter 5, Configuring Ethernet Interfaces*.

Supported Modules and Encapsulations

You can enable hardware multicast packet replication on port 8 of the following high-density Ethernet modules:

- GE-8 I/O module (pairs with the GE-HDE line module)
- ES2-S1 GE-8 IOA (pairs with the ES2 4G LM and the ES2 10G LM)

When enabled, the hardware multicast packet replication feature defines the encapsulation of the egress multicast packet. The following encapsulations are supported:

- IPv4 over Gigabit Ethernet
- IPv4 over VLAN
- IPv4 over S-VLAN

NOTE: 802.3ad link aggregation group (LAG) bundles do not support hardware multicast packet replication.

The hardware multicast packet replication feature also provides an interface over which you can configure the following:

- IP MTU
- Ethernet MTU
- Egress IP policy
- Egress VLAN policy
- QoS

Relationship with OIF Mapping

Multicast OIF mapping enables the router to decrease the inefficiencies associated with replicating streams of multicast traffic. Using OIF maps, IGMP joins that the router receives on a subscriber interface can be mapped to a dedicated multicast VLAN.

The hardware multicast packet replication feature enables you to redirect each of the IP interfaces on a line module over a dedicated multicast VLAN to a single IP interface over port 8. The FC is only required to send a single packet per dedicated multicast VLAN to the I/O module or IOA. The module then replicates this packet to the appropriate ports.

For more information about configuring OIF mapping, see *Configuring Group Outgoing Interface Mapping* in *Chapter 2, Configuring IGMP*.

Hardware Multicast Packet Replication Considerations

When configuring hardware multicast packet replication, the following considerations apply.

- Do not configure or transmit routing protocols over port 8. The FC drops traffic routed to an IP interface stacked over port 8.
- We recommend that you configure the IP address of the IP interface over port 8 to be unnumbered.
- We recommend that you configure an IP interface over a VLAN over one of the physical ports to reference the IP interface over the same VLAN over port 8.

  You cannot create the following configurations:

  - When two IP interfaces configured over a port reference the same IP interface over port 8. The system does not accept this configuration attempt because you typically configure the hardware multicast packet replication feature to redirect multicast traffic over one VLAN, then redirect it to the same VLAN on port 8.
  - When the IP interface configured with the hardware multicast packet replication attribute is not installed on a line module that supports hardware multicast packet replication.
  - When the IP interface designated by the hardware multicast packet replication attribute is not installed on a line module that supports hardware multicast packet replication.
  - When the IP interface designated by the hardware multicast packet replication attribute is not on the same line module as the IP interface configured with this attribute.
- When you configure a unique source MAC address for VLANs on port 8, the hardware multicast packet replication hardware stamps the source MAC address on the VLAN, overwriting any MAC address that you configured. For more information, see *JUNOSe Physical Layer Configuration Guide, Chapter 5, Configuring Ethernet Interfaces*.

- The regular multicast implementation utilizes interface stacking that provides a unique IP attachment point for each elaboration of the egress multicast packet.

  For the hardware multicast packet replication feature, you must attach policies to an interface stack over port 8 that defines the encapsulation of the egress multicast traffic. The system supports policies over port 8 just as it is above any of the other ports on this line module.

  Policies applied to the interface stack over port 8 affect the packets traversing this stack whether or not the packet is destined for one port or all of the physical ports. Therefore, you cannot apply different egress policies to multicast traffic for the interfaces stacked above different ports, or rate limit on an individual interface over a port. You also cannot monitor policy statistics on individual interfaces over a port.

  Instead, you can apply egress policy to an interface stacked over port 8. The system applies the policy before the packet has been elaborated for each of the ports.

- The JUNOSe QoS component provides hierarchical egress scheduling and shaping on Gigabit Ethernet ports 0–7. The regular multicast implementation replicates packets on the FC, with each replicated packet placed on a line module queue destined for a single physical port. The line module queue can also receive QoS behavior specific to that queue.

  For the hardware multicast packet replication feature, the FC does not replicate the packet for each of the individual ports. Instead, it places the packet on a special queue destined for port 8.

  You can configure QoS on the packets flowing through port 8, but this has limited value because each packet passed through this port can be transmitted through one of more of the physical ports. Therefore, the packets placed on this special queue might not receive the same QoS behavior as ports 0–7.

  We recommend that you configure the network so the I/O or IOA queues are not oversubscribed. The traffic transmitted by the physical port is a combination of packets from the two I/O or IOA queues. When the sum of the packets in these queues is greater than line rate, the system can drop traffic that is not using hardware multicast packet replication.

  When you configure a traffic shaper on a physical port and configure hardware multicast packet replication, the packets created using the feature avoid the traffic shaper for that port. To control this, you can use traffic shaper on the physical port and port 8. The sum of the traffic shapers must be less than or equal to the line rate of the port.

  A traffic shaper on port 8 can result in the overall utilization of egress bandwidth for any one port being less the line rate because the packets being replicated might not be transmitted to every port. Packets destined to some of the ports contribute to the traffic shaping for all of the ports on the I/O module or IOA.

Configuring Hardware Multicast Packet Replication

To configure hardware multicast packet replication:

1. Configure port 8 on a high-density Ethernet module to accept redirected egress multicast traffic.

   a. Specify the Gigabit Ethernet interface on port 8.

   b. Create a VLAN major interface.

   c. Create a VLAN subinterface.

   d. Assign a VLAN ID.

   e. Configure an unnumbered IP interface.

   f. Enable IGMP on the interface with only multicast-data-forwarding capability.

   ```
   host1(config)#interface gigabitEthernet 2/8
   host1(config-if)#encapsulation vlan
   host1(config-if)#interface gigabitEthernet 2/8.1
   host1(config-if)#vlan id 1
   host1(config-if)#ip unnumbered loopback 0
   host1(config-if)#ip igmp version passive
   ```

2. Configure an IP interface to redirect egress multicast traffic to port 8.

   a. Create a VLAN subinterface.

   b. Assign a VLAN ID.

   c. Assign an IP address.

   d. Configure the interface to redirect egress multicast traffic to port 8.

   ```
   host1(config)#interface gigabitEthernet 2/0.101
   host1(config-if)#vlan id 1
   host1(config-if)#ip address 10.1.1.1 255.255.255.0
   host1(config-if)#ip multicast ioa-packet-replication gigabitEthernet 2/8.1
   ```

encapsulation vlan

- Use to configure VLAN as the encapsulation method for the interface.
- Example

  ```
  host1(config-if)#encapsulation vlan
  ```

- Use the **no** version to disable VLAN on an interface.

ip igmp version

- Use to set the IGMP version (1, 2, or 3) for the interface or specify a passive interface with only multicast-data-forwarding capability (passive).
- Example

  host1:boston(config-if)#**ip igmp version passive**

- Use the **no** version to set the version to the default, IGMPv2.

ip multicast ioa-packet-replication

- Use to configure hardware multicast packet replication on port 8 of a high-density Ethernet module.
- Example

  host1(config-if)#**ip multicast ioa-packet-replication gigabitEthernet 3/8.1**

- Use the **no** version to disable hardware multicast packet replication.

ip unnumbered

- Use to configure an unnumbered IP interface.
- This command enables IP processing on an interface without assigning an explicit IP address to the interface.
- You must specify an interface location, which is the identifier of another interface on which the router has an assigned IP address. This interface cannot be another unnumbered interface.
- Example

  host1(config-if)#**ip unnumbered loopback 10**

- Use the **no** version to disable IP processing on the interface.

Monitoring Hardware Multicast Packet Replication

This section describes how to monitor hardware multicast packet replication.

Port Statistics

Use the **show interfaces gigabitEthernet** command to display port statistics for port 8. For port 8, queue statistics have no direct relationship to any of the 8 ports because each packet transmitting through the queue can be sent through 1 or more of the 8 physical ports. For more information, see *Monitoring Ethernet Interfaces* in *JUNOSe Physical Layer Configuration Guide, Chapter 5, Configuring Ethernet Interfaces*.

IP and VLAN Statistics

Use the **show vlan subinterface** command to display statistics for a VLAN interface configured over port 8. For more information, see *Monitoring Ethernet Interfaces* in *JUNOSe Physical Layer Configuration Guide, Chapter 5, Configuring Ethernet Interfaces*.

Use the **show ip interface** command to display statistics for an IP interface configured over port 8. For more information, see *Monitoring IP* in *JUNOSe IP, IPv6, and IGP Configuration Guide, Chapter 1, Configuring IP*.

Multicast traffic redirected by the hardware multicast packet replication feature is displayed in the statistics for the IP or VLAN interface over port 8, not the original IP or VLAN interface over the physical port.

The statistics for the IP or VLAN interface over port 8 reflect the number of packets that passed through this interface destined for the hardware multicast packet replication hardware. These statistics have no direct correlation to the number of packets being transmitted from any of the physical ports.

IGMP Statistics

Use the **show ip igmp interface** command to display statistics, including hardware multicast packet replication configuration, for an IP interface stacked over port 8. For more information, see *Monitoring IGMP* in *Chapter 2, Configuring IGMP*.

Blocking and Limiting Multicast Traffic

You can either block mroute creation, limit the multicast bandwidth admitted on an outgoing interface, or limit outgoing interface creation on a port.

Blocking Mroutes

By default, when an interface that is configured with one or more multicast protocols (for example, PIM or IGMP) receives multicast traffic, even when the scope of that traffic exceeds link-local, the virtual router creates an mroute. You can use the **ip block-multicast-sources** command to block all multicast traffic with a scope larger than link-local (for example, global) and prevent mroute creation under these conditions.

NOTE: Issuing this command does not affect reception of link-local multicast packets.

ip block-multicast-sources

- Use to prevent mroute creation by blocking multicast traffic that has a scope larger than link-local (for example, global).
- Example

  host1(config-if)#**ip block-multicast-sources**

- Use the **no** version to restore the default behavior of creating mroutes on received multicast packets.

Limiting Interface Admission Bandwidth

Interface-level multicast admission control is performed when an OIF on the interface is added to the mroute for a given (S,G) multicast data stream and the multicast bandwidth map contains a **set admission-bandwidth** action for that (S,G).

When enabled, the admission-bandwidth for a particular (S,G) is read from the multicast bandwidth map and recorded in the mroute when the (S,G) mroute is created. When an OIF is subsequently added to the mroute, the OIF is blocked from forwarding data if the additional bandwidth contributed by the (S,G) would exceed the admission-bandwidth limit for the interface.

CAUTION: Before you can limit interface-level admission bandwidth, you must first create a bandwidth map. See *Defining a Multicast Bandwidth Map* on page 11 for details.

Enabling Interface Admission Bandwidth Limitation

You can use the **ip multicast admission-bandwidth-limit** command to enable multicast admission control on interfaces (including dynamic IP interfaces) that are configured to run IGMP. You can also use this command on a PIM (sparse-mode, dense-mode, or sparse-dense-mode) interface if IGMP is configured on the interface (including the **ip igmp version passive** command).

ip multicast admission-bandwidth-limit

- Use to limit bandwidth for an interface that accepts IGMP groups.
- Example

  host1:boston(config-if)#**ip multicast admission-bandwidth-limit 2000000**

- Use the **no** version to remove the bandwidth limitation for the interface.

OIF Interface Reevaluation Example

If you change the admission bandwidth for an interface, all mroutes with that interface as an OIF are reevaluated as follows:

- If the bandwidth limit is increased, blocked OIFs may become unblocked. If the interface is a blocked OIF on multiple mroutes, the order in which the mroutes are visited, and which (S,G) streams become unblocked, is not specified.
- If the bandwidth limit is decreased, no currently admitted OIFs are blocked. However, no new OIFs are admitted until the total admitted bandwidth for the interface drops below the new limit.

- If the bandwidth is increased to the point that the bandwidth limit for an interface is now exceeded, no currently admitted OIFs for the affected mroutes are blocked. However, no new OIFs are admitted until the total admitted bandwidth drops below the configured limit.

NOTE: If the multicast bandwidth map that includes the **set admission-bandwidth command** is changed, all affected mroutes are reevaluated in the same manner described previously.

As an example of this function, if the interface has accepted a total bandwidth of 2000000 bps, and you set a limit of 1000000 bps on the interface, the router does not disconnect any already connected OIFs but prevents the interfaces from accepting any more groups. Over time, some groups leave the interfaces and, eventually, the interface limit of 1000000 bps is reached and maintained by the router.

If you set limits for both a port and interfaces on that port, the router uses the lower of the two limits when determining whether or not an interface can accept any new IGMP groups. For example, if you specify an admission bandwidth limit of 2000000 bps for the port and 3000000 bps groups for each interface, additional groups can only be accepted until the port limit of 2000000 bps is reached.

Creating Mroute Port Limits

When a multicast forwarding entry (that is, an mroute) is added with an outgoing interface (OIF) on a port, the OIF count for that port is incremented. If you configure a port limit, and the OIF count on the port exceeds that limit, no OIFs on that port are added to mroutes (that is, OIFs are blocked).

mroute port limit

- Use to configure a limit on the number of mroute OIFs that can be added across different virtual routers, on a port.
- Example

  host1(config)#**mroute port 3/0 limit 10**

- Use the **no** version to remove any OIF port limits.

Limiting Port Admission Bandwidth

Port-level multicast admission control is performed when an OIF on that port is added to the mroute for a given (S,G) multicast data stream and the multicast bandwidth map contains a **set admission-bandwidth** action for that (S,G).

When enabled, the admission-bandwidth for a particular (S,G) is read from the multicast bandwidth map and recorded in the mroute when the (S,G) mroute is created. If you configure a port limit and the OIF count on the port exceeds that limit, no OIFs on that port are added to mroutes (that is, OIFs are blocked).

When a multicast forwarding entry (an mroute) is added with an outgoing interface, OIF is blocked from forwarding data if the additional bandwidth contributed by the (S,G) would exceed the admission-bandwidth limit for the port on which the interface resides.

CAUTION: Before you can limit port-level admission bandwidth, you must first create a bandwidth map. See *Defining a Multicast Bandwidth Map* on page 11 for details.

Enabling Port Admission Bandwidth Control

You can use the **mroute port admission-bandwidth-limit** command to limit the total multicast bandwidth that can be admitted on a port. The admitted bandwidth is summed across all virtual routers with IPv4 and IPv6 mroutes that have OIFs on the port.

NOTE: Admission bandwidth values for a given (S,G) mroute are determined from the bandwidth map. See *Defining a Multicast Bandwidth Map* on page 11 for details.

Dynamic Port Admission Bandwidth Control

You can configure the system to dynamically limit the total multicast bandwidth that can be admitted on a port. The system performs dynamic port-level admission control when an OIF on that port is added to the mroute for a given <S, G> multicast stream.

After the priority bandwidth limit on the port is reached, OIFs on the prioritized <S, G> are only allowed to forward the traffic and unprioritized <S, G> streams are blocked from forwarding data on the OIF.

To enable a priority value for the <S, G> multicast stream, issue the **set priority** command in the multicast bandwidth map. A priority value of 0 indicates an unprioritized stream and any value other than 0 indicates a prioritized stream. Currently there is no support for classification of prioritized streams. For more information about the **set priority** command, see *Defining a Multicast Bandwidth Map* on page 11.

You can configure limits for the bandwidth that is dynamically admitted on the port. The priority bandwidth limit controls the priority bandwidth admitted on a port. The hysteresis limit sets the minimum priority bandwidth limit before the system evaluates mroutes and admits any blocked OIFs.

mroute port admission-bandwidth-limit

- Use to configure a limit on the total multicast bandwidth that can be admitted on a port.
- Use the **priority-bandwidth-limit** keyword to configure the priority bandwidth admitted on a port.
- Use the **hysteresis** keyword to configure the minimum priority bandwidth limit before the system evaluates mroutes and admits any blocked OIFs.
- Example

  host1(config)#**mroute port admission-bandwidth-limit 3000000**

- Use the **no** version to remove any OIF admission bandwidth limits.

OIF Port Reevaluation Example

If you change the admission bandwidth for a port, all mroutes with an OIF on that port are reevaluated as follows:

- If the bandwidth limit is increased, blocked OIFs can become unblocked. However, the order in which the mroutes are visited, and which (S,G) streams become unblocked, is not specified.
- If the bandwidth limit of a port is decreased, no currently admitted OIFs are blocked. However, no new OIFs are admitted until the total admitted bandwidth for the port drops below the new limit.
- If the bandwidth is increased to the point that the bandwidth limit for an interface is now exceeded, no currently admitted OIFs for the affected mroutes are blocked. However, no new OIFs are admitted until the total admitted bandwidth drops below the configured limit.

NOTE: If the multicast bandwidth map that includes the **set admission-bandwidth command** is changed, all affected mroutes are reevaluated in the same manner described previously.

As an example of this function, if the port has accepted a total bandwidth of 3000000 bps, and you set a limit of 2000000 bps on the port, the router does not disconnect any already connected OIFs but prevents the interfaces from accepting any more groups. Over time, some groups leave the interfaces and, eventually, the port limit of 2000000 bps is reached and maintained by the router.

If you set limits for both a port and interfaces on that port, the router uses the lower of the two limits when determining whether or not an interface can accept any new IGMP groups. For example, if you specify an admission bandwidth limit of 2000000 bps for the port and 3000000 bps groups for each interface, additional groups can only be accepted until the port limit of 2000000 bps is reached.

Deleting Multicast Forwarding Entries

You can clear one or more forwarding entries from the multicast routing table. However, if you do so, the entries might reappear in the routing table if they are rediscovered.

clear ip mroute

- Use to delete IPv4 multicast forwarding entries.
- If you specify an *, the router clears all IP multicast forwarding entries.
- If you specify the IPv4 address of a multicast group, the router clears all multicast forwarding entries for that group.
- If you specify the IPv4 address of a multicast group and the IPv4 address of a multicast source, the router clears the multicast forwarding entry that matches that group and source.
- Example

  host1:boston#**clear ip mroute ***

- There is no **no** version.

Monitoring IP Multicast Settings

To display general information about the IP multicast configuration on the router, use the following **show** commands.

show ip mroute

- Use to display information about all or specified multicast forwarding entries.
- Specify a multicast group IP address or both a multicast group IP address and a multicast source IP address to display information about particular multicast forwarding entries.
- Use the **summary** option to see a summary rather than a detailed description.
- Use the **count** option to display the number of multicast forwarding entries.
- Use the **statistics** option to display statistics for packets received through all multicast forwarding entries that the router has added to the multicast routing table and established on the appropriate line modules.
- Use the **active** option to display the active multicast routes with admission bandwidth greater than the specified bandwidth threshold. The default is 4000 bps.
- Field descriptions
  - (S, G)—IP addresses of the multicast source and the multicast group
  - Admission bandwidth—Admission bandwidth per mroute, in bps
  - QoS bandwidth—QoS bandwidth per mroute, in bps
  - Uptime—Length of time that the (S,G) pair has been active, in *days hours:minutes:seconds* format
  - Data Rate—Flow rate for the threshold entry, in Kbps

- SPT Threshold—SPT threshold value for the entry, in Kbps
- Threshold—Threshold value for the entry, in Kpbs
- Expires—Length of time that the (S,G) pair can be active, in *days hours:minutes:seconds* format or *never*
- RPF route—IP address and subnetwork mask of the RPF route
- incoming interface—Type and specifier of the incoming interface for the RPF route
- neighbor address—IP address of the neighbor
- State/Owner—Owner of the route
  - Local—Route belonging to the local interface
  - Static—Static route
  - Other protocols—Route established by a protocol such as RIP or OSPF
- Incoming interface list—List of incoming interfaces on the router. Details include:
  - Type of interface and its specifier
  - Action that the interface takes with packets: Accept or Discard
  - Multicast protocol that owns the interface
- Outgoing interface list—List of outgoing interfaces on the router. Details include:
  - Type of interface and its specifier
  - Action that the interface takes with packets: Forward or Blocked (port-limit)
  - Protocol running on the interface: PIM, DVMRP, or IGMP
  - Amount of time that the interface has been active in this multicast forwarding entry, in *days hours:minutes:seconds* format
  - Length of time that the interface can remain active in this multicast forwarding entry, in *days hours:minutes:seconds* format or *never*
- Counts—Number of types of source group mappings
  - (S, G)—Number of (S, G) entries
  - (*, G)—Number of (*, G) entries

- Example 1—Constant bandwidth bit rate

```
host1#show ip mroute
IP Multicast Routing Table

(S, G) uptime d h:m:s
  [Data rate: Kbps]  [SPT Threshold: Kbps]  [Threshold: Kbps]
  [Admission bandwidth: bps]
  [QoS bandwidth: bps]
  RPF route: addr/mask, incoming interface
             neighbor address, owner route-owner
  Incoming interface list:
      Interface (addr/mask), State/Owner [(RPF IIF)]
  Outgoing interface list:
      Interface (addr/mask), State/Owner, Uptime/Expires
```

```
(10.0.10.1, 225.1.1.1) uptime 0 00:10:31
  Data rate: 2132 Kbps, Threshold 500 Kbps
  Admission bandwidth: 2000000 bps
  RPF route: 10.0.10.0/24, incoming interface atm5/3.1010
             neighbor 10.0.10.8, owner Local
  Incoming interface list:
      atm5/3.1010 (10.0.10.8/24), Accept/Pim (RPF IIF)
  Outgoing interface list:
      atm5/1.108 (108.0.8.5/8), Forward/Pim, 0 00:02:52/never
      atm5/1.109 (107.0.8.4/8), Forward/Pim, 0 00:10:07/never

(1.1.1.1, 225.1.1.1) uptime 0 00:00:34, never expires
RPF route: 1.0.0.0/8, incoming interface ATM5/1.200
             neighbor 2.2.2.2, owner Netmgmt
  Incoming interface list:
      ATM5/1.200 (2.1.1.1/8), Accept/Igmp (RPF IIF)
  Outgoing interface list:
      ATM5/1.300 (3.1.1.1/8), Forward/Igmp, 0 00:00:34/never

Counts:     2 (S, G) entries
            0 (*, G) entries
```

NOTE: The (S,G) entry (1.1.1.1, 225.1.1.1) is the permanent mroute.

- Example 2—Adaptive bandwidths enabled

```
Host1#show ip mroute
                  IP Multicast Routing Table

(S, G) uptime d h:m:s[, expires d h:m:s]
  [Admission bandwidth: bps]
  [QoS bandwidth: bps]
  RPF route: addr/mask, incoming interface
             neighbor address, owner route-owner
  Incoming interface list:
      Interface (addr/mask), State/Owner [(RPF IIF)]
  Outgoing interface list:
      Interface (addr/mask), State/Owner, Uptime/Expires

 (10.0.1.9, 225.1.1.1) uptime 0 00:00:23
  Admission bandwidth: 1998000 bps (adaptive)
  QoS bandwidth: 1998000 bps (adaptive)
  RPF route: 10.0.0.0/8, incoming interface ATM2/1.200
             neighbor 21.1.1.1, owner Netmgmt
  Incoming interface list:
      ATM2/1.200 (21.2.2.2/8), Accept/Pim (RPF IIF)
  Outgoing interface list:
      ATM2/1.300 (31.2.2.2/8), Blocked (port-adm-limit)/Pim, 0
00:00:23/never

Counts: 1 (S, G) entries
        0 (*, G) entries
```

show ip mroute active

- Use to display the active multicast routes with admission bandwidth greater than the specified bandwidth threshold.
- The default is 4000 bps.
- Field descriptions
  - See the **show ip mroute** command and the **show ip mroute summary** command for descriptions of all fields.
- Example 1—Displays active multicast routes with bandwidth above 10000 bps

```
host1#show ip mroute active 10000
              Active IP Multicast Routes >=10000 bps

(S, G) uptime d h:m:s[, expires d h:m:s]
  [Admission bandwidth: bps]
  [QoS bandwidth: bps]
  RPF route: addr/mask, incoming interface
             neighbor address, owner route-owner
  Incoming interface list:
      Interface (addr/mask), State/Owner [(RPF IIF)]
  Outgoing interface list:
      Interface (addr/mask), State/Owner, Uptime/Expires

(52.0.0.1, 232.0.0.1) uptime 0 00:01:07
  Admission bandwidth: 47000 bps (adaptive)
  QoS bandwidth: 47000 bps (adaptive)
  RPF route: 52.0.0.0/24, incoming interface ATM2/1.17
      neighbor 17.0.0.2, owner NetmgmtRpf
  Incoming interface list:
      ATM2/1.17 (17.0.0.2/24), Accept/Igmp (RPF IIF)
  Outgoing interface list:
      NULL

Counts: 1 (S, G) entries
        0 (*, G) entries
```

- Example 2—Displays the summary of active multicast routes

```
host1#show ip mroute summary active
              Active IP Multicast Routes >=4000 bps

Group Address   Source Address  RPF route          RPF Iif         #Oifs
--------------- --------------- ------------------ --------------- -----
232.0.0.1       51.0.0.1        51.0.0.0/24        ATM3/1.17       0
232.0.0.2       51.0.0.1        51.0.0.0/24        ATM3/1.17       0
232.0.0.3       51.0.0.1        51.0.0.0/24        ATM3/1.17       0
Counts: 3 (S, G) entries
        0 (*, G) entries
```

show ip mroute count

- Use to display information about the number of groups and sources.
- Specify a multicast group address or both a multicast group address and a multicast source address to display information about a particular multicast forwarding entry.
- Field descriptions
  - Counts—Number of types of source group mappings
    - (S, G)—Number of (S,G) entries
    - (*, G)—Number of (*,G) entries
- Example

```
host1#show ip mroute count
                IP Multicast Routing Table

Counts:     2 (S, G) entries
            0 (*, G) entries
```

show ip mroute statistics

- Use to display statistics for packets received through multicast routes that the router has added to the multicast routing table and established on the appropriate line modules.
- Specify a multicast group IP address or both a multicast group IP address and a multicast source IP address to display information about a particular multicast forwarding entry.
- Field descriptions
  - See the **show ip mroute** command for descriptions of all fields except the Statistics field.
  - Statistics

NOTE: The display shows statistics after the VR has added the multicast route to the multicast routing table and established the route on the appropriate line module. Statistics for interactions that take place before the route is established on the line module are not displayed.

    - Received—Number of packets and bytes that the VR received for this multicast route
    - Forwarded—Number of packets and bytes that the VR has forwarded for this multicast route
    - Rcvd on OIF—Number of packets that the VR has received on the outgoing interface (OIF) for this multicast route
- Example

```
host1#show ip mroute statistics
IP Multicast Routing Table

(S, G) uptime d h:m:s[, expires d h:m:s]
  [Admission bandwidth: bps]
  [QoS bandwidth: bps]
  RPF route: addr/mask, incoming interface
             neighbor address, owner route-owner
```

```
  Incoming interface list:
      Interface (addr/mask), State/Owner [(RPF IIF)]
  Outgoing interface list:
      Interface (addr/mask), State/Owner, Uptime/Expires
(10.0.1.9, 225.1.1.1) uptime 0 00:00:23
  Admission bandwidth: 2000000 bps
  QoS bandwidth: 2000000 bps
  RPF route: 10.0.0.0/8, incoming interface ATM2/1.200
             neighbor 21.1.1.1, owner Netmgmt
  Incoming interface list:
      ATM2/1.200 (21.2.2.2/8), Accept/Pim (RPF IIF)
  Outgoing interface list:
      ATM2/1.300 (31.2.2.2/8), Blocked (port-adm-limit)/Pim, 0
00:00:23/never
  Statistics:
      Received   : 23 pkts, 1472 bytes
      Forwarded  : 0 pkts, 0 bytes
      Rcvd on OIF: 0 pkts

Counts: 1 (S, G) entries
        0 (*, G) entries
```

show ip mroute summary

- Use to display a summary of all or specified multicast routes.
- Specify a multicast group IP address or both a multicast group IP address and a multicast source IP address to display information about a particular multicast forwarding entry.
- Field descriptions
  - Group Address—IP address of the multicast group
  - Source Address—IP address of the multicast source
  - RPF route—IP address and network mask of the RPF route
  - RPF Iif —Type and identifier for the incoming interface for the RPF route
  - #Oifs—Number of outgoing interfaces
  - Counts—Numbers of types of (S,G) pairs
    - (S,G)—Number of (S,G) entries
    - (*,G)—Number of (*,G) entries
- Example

```
host1#show ip mroute summary
                 IP Multicast Routing Table

Group Address   Source Address  RPF route    RPF Iif         #Oifs
--------------- --------------- ---------    -------------   ----
224.0.1.39      52.1.1.1        51.1.1.1/32  Register IIF    0
224.0.1.40      51.1.1.1        51.1.1.1/32  loopback1       1

Counts:     2 (S, G) entries
            0 (*, G) entries
```

show ip multicast protocols

- Use to display information about multicast protocols enabled on the router.
- Use the **brief** option to display a summary of information rather than a detailed description.
- Field descriptions
  - Multicast Protocols—Multicast protocols on this router
  - Protocol—Name of the multicast protocol
  - Type—Mode of the multicast protocol
    - For DVMRP—Dense
    - For PIM—Sparse, Dense, or Sparse-Dense
    - For IGMP—Local
  - Interfaces
    - registered—Number of interfaces on which the protocol is configured
    - owned—Number of interfaces that a protocol owns. If you configure only IGMP on an interface, IGMP owns the interface. However, if you configure IGMP and either PIM or DVMRP on the same interface, PIM or DVMRP owns the interface.
  - Registered interfaces—Information about interfaces on which the protocol is configured:
    - Types and specifiers of interfaces. For details about interface types and specifiers, see *JUNOSe Command Reference Guide, About This Guide*.
    - Protocols configured on the interface and the protocol that owns the interface. If you configure only IGMP on an interface, IGMP owns the interface. However, if you configure IGMP and PIM or DVMRP on the same interface, PIM or DVMRP owns the interface.
- Admission-bandwidth—Actual admission bandwidth/configured admission bandwidth (in bps)
- QoS Adjust—Bandwidth of QoS adjustment, in bps
  - Count—Number of multicast protocols on the VR
  - Active <S,G> count—Number of active S,G data streams on the interface
  - Blocked <S,G> count—Number of blocked S,G data streams on the interface
- Example

```
host1#show ip multicast protocols
Multicast protocols:

Protocol Pim
    Type: Sparse
    Interfaces: 1 registered, 1 owned
    Registered interfaces:
        ATM2/1.103 (103.0.0.2/24) owner Pim

Protocol Igmp
    Type: Local
    Interfaces: 1000 registered, 1000 owned
```

```
    Registered interfaces:
        ATM2/0.131 (13.0.0.1/24) local Igmp owner Igmp
            Admission-bandwidth 2000000/10000000 bps
            QoS Adjust 2000000 bps
        Active <S,G> count   15
        Blocked <S,G> count  10

        ATM2/0.132 (13.0.0.2/24) local Igmp owner Igmp
            Admission-bandwidth 0/10000000 bps
            QoS Adjust 0 bps
        Active <S,G> count   25
        Blocked <S,G> count  10

        ATM2/0.133 (13.0.0.3/24) local Igmp owner Igmp
            Admission-bandwidth 8000000/10000000 bps
            QoS Adjust 0 bps
...
Count: 2  protocols
```

show ip multicast protocols brief

- Use to display a summary of information about multicast protocols enabled on the router.
- Field descriptions
  - Protocol—Name of the multicast protocol
  - Registered Interfaces—Number of interfaces on which the protocol is configured
  - Owned Interfaces—Number of interfaces that a protocol owns. If you configure only IGMP on an interface, IGMP owns the interface. However, if you configure IGMP and either PIM or DVMRP on the same interface, PIM or DVMRP owns the interface.
  - Type—Mode of the multicast protocol
    - For DVMRP—Dense
    - For PIM—Sparse, dense, or sparse-dense
    - For IGMP—Local
  - Count—Number of multicast protocols on the VR
- Example

```
host1#show ip multicast protocols brief

Protocol  Registered  Owned       Type
          Interfaces  Interfaces
--------- ----------  ----------  -------------------
Pim       2           2           Sparse Dense
Igmp      1           0           Local

Count: 2 protocols
```

show ip multicast routing

- Use to display information about the status of IP multicast on the VR.
- Example

```
host1#show ip multicast routing
Multicast forwarding is enabled on this router
Multicast graceful restart is complete (timer 0 seconds) on this router
Multicast cache-miss processing is enabled on this router
```

show mroute port count

- Use to display the mroute port outgoing interface, limits, counts, bandwidth settings, and bandwidth accepted.

NOTE: This command displays information for mroutes on a port across all virtual routers.

- Field descriptions
  - Port—Slot/port value on the router
  - Limit—Port limit value defined for the specified port; -l indicates that no mrout port limits have been configured for the port
  - Count—Number of mroute outgoing interfaces on the specified port
  - BW bps—Bandwidth limit, in bits per second
  - Priority BW bps—Priority bandwidth limit, in bits per second
  - Admitted—Bandwidth admitted on the port, in bits per second
- Example

```
host1#show mroute port count

BW     Priority
Port    Limit   Count    bps     BW bps    Hysteresis   Admitted
-----   -----   -----   -----   --------   ----------   --------
1/1/0   None    1       None    None       85           0
1/1/1   None    2       15000   10000      85           2000
```

Support for Multicast Router Information

When you enable multicast routing on a virtual router, the router acts as a multicast router information (mrinfo) server. This feature enables the router to respond to mrinfo requests from other network hosts. Specifically, E-series virtual routers respond to DVMRP ask neighbors and DVMRP ask neighbors2 requests.

Each virtual router responds to mrinfo requests with a list of multicast interfaces and their IP addresses. If appropriate, the virtual router also supplies the following information for each interface:

- Current functional status of the interface (for example, if the interface is down).
- Information as to whether the interface is disabled and the reason for the interface being disabled—either because IP is not configured on the interface or because the interface has been disabled through the software.
- Whether the interface is performing the IGMP queries for this subnet.
- Information about PIM neighbors:

  If PIM is configured on the interface, the virtual router supplies a list of the interface's PIM neighbors and indicates which neighbors are leaf neighbors.
- Information about DVMRP and GRE tunnels:

  If the interface is an endpoint of a tunnel, the virtual router specifies the IP address of the endpoint of the tunnel.

BGP Multicasting

BGP multicasting (MBGP) is an extension of the BGP unicast routing protocol. Many of the functions available for BGP unicasting are also available for MBGP.

The MBGP extensions specify that BGP can exchange information within different types of *address families*. The address families available are unicast IPv4, multicast IPv4, and VPN-IPv4. When you enable BGP, the router employs unicast IPv4 addresses by default.

We recommend you be thoroughly familiar with BGP before configuring MBGP. See *JUNOSe BGP and MPLS Configuration Guide, Chapter 1, Configuring BGP Routing,* for detailed information about BGP and MBGP.

Investigating Multicast Routes

You can use the **mtrace** command to trace the path that multicast packets take from a source to a destination through a multicast group address. This command is similar to the **traceroute** command for investigating unicast routes.

mtrace

- Use to trace the path that multicast packets take to a destination.
- Specify the unicast IP address of the source for the packets.
- To direct the packets to a particular destination, specify the unicast address for that destination. If you do not specify a destination, the router traces the route from the device on which you issue the command.
- To direct the packets through a particular multicast group address, specify that multicast group address. If you do not specify a multicast group address, the router traces the route through the MBone audio multicast group.
- To send the trace to a particular device, specify the IP address of that device. If you do not specify a response address, the router sends the trace to an IP address on the router.
- To investigate a problem at a particular point in the route, specify the maximum number of hops for the trace. The default number of hops is 64.
- The trace starts at the destination and works back to the source.
- Field descriptions
  - Tracing multicast route from *a.a.a.a* to *b.b.b.b* for group *c.c.c.c* using response address *d.d.d.d*—A description of the trace is as follows:
    - *a.a.a.a*—IP address of the source
    - *b.b.b.b*—IP address of the destination
    - *c.c.c.c*—IP address of the multicast group
    - *d.d.d.d*—IP address of the router to which the router sends the trace
  - Received mtrace response packet of length *n*—Length of the response packet, in bytes
  - Each line of the trace has the following format: *hops. ip-address Protocol: protocol FwdingCode:forwarding code*
    - *hops*—Number of hops from the destination to this intermediate router
    - *ip-address*—IP address of the intermediate router
    - *protocol*—Multicast protocol running on the intermediate router. A value of 12 indicates IGMP; other values comply with *A "traceroute" Facility for IP Multicast – draft-ietf-idmr-traceroute-ipm-07.txt*.
    - *FwdingCode*—Forwarding information or error associated with this hop. For example, RPF iif indicates that the request arrived on the expected RPF interface for this source group. For more information about the forwarding information or error codes, see *A "traceroute" Facility for IP Multicast – draft-ietf-idmr-traceroute-ipm-07.txt*.

- Example

  host1#**mtrace 100.4.4.4 40.1.1.1 232.1.1.1**
  Tracing multicast route from 100.4.4.4 to 40.1.1.1 for group 232.1.1.1 using response address 10.6.129.56
  (Press ^c to stop.)
  Received mtrace response packet of length 88
  1. 40.1.1.1 Protocol: PIM(3) FwdingCode: RPF iif(9)
  2. 21.2.2.2 Protocol: PIM(3) FwdingCode: Reached RP(8)

- There is no **no** version.

Chapter 2

Configuring IGMP

IP hosts use Internet Group Management Protocol (IGMP) in IPv4 to report their multicast group memberships to neighboring routers. Similarly, multicast routers, such as an E-series router, use IGMP to discover which of their hosts belong to multicast groups.

This chapter describes how to configure IGMP for IP multicast on an E-series router; it contains the following sections:

IGMP Overview

The IPv4 address scheme assigns class D addresses for IP multicast. IGMP is the protocol that uses these addresses, which can be in the range 224.0.0.0 to 239.255.255.255. The following addresses have specific functions or are unavailable:

- 224.0.0.0 is reserved—you cannot assign it to a group.
- 224.0.0.1 is the all-hosts address—a packet sent to this address reaches all hosts on a subnet.
- 224.0.0.2 is the all-routers address—a packet sent to this address reaches all routers on a subnet.

This implementation of IGMP complies with IGMP versions 1, 2, and 3. IGMPv3 supports source-specific join and leave messages and is backward compatible with IGMPv1 and IGMPv2.

IGMPv2 mode interfaces exchange the following types of messages between routers and hosts:

- Group membership queries
- Group membership reports
- Leave group membership messages

IGMPv3 mode interfaces exchange the following types of messages with IGMPv3 hosts:

- Group membership queries
- IGMPv3 group membership reports

Group Membership Queries

A multicast router can be a querier or a nonquerier. Only one querier is on a network at any time. Multicast routers monitor queries from other multicast routers to determine the status of the querier. If the querier detects a query from a router with a lower IP address, it relinquishes its role to that router.

IGMPv1 and IGMPv2 mode interfaces send two types of group membership queries to hosts on the network:

- General queries to the all-hosts group address (224.0.0.1)
- Specific queries to the appropriate multicast group address

IGMPv3 mode interfaces send the following types of queries to IGMPv3 hosts:

- General queries
- Group-specific queries
- Source-specific queries

The purpose of a group membership query is to discover the multicast groups to which a host belongs.

IGMPv2 and IGMPv3 group membership queries have a Max Response Time field. This response time is the maximum amount of time that a host can take to reply to a query.

Group Membership Reports

When a host receives a group membership query, it identifies the groups associated with the query and determines to which groups the query belongs. The host then sets a timer, with a value less than the Max Response Time field in the query, for each group to which it belongs.

When the timer expires, the host sends a group membership report to the group address. When a multicast router receives a report, it adds the group to the membership list for the network and sets a timer to the *group membership interval*. The router calculates the group membership interval using the following formula of configurable IGMP values:

(query interval x robustness value) + query maximum response time

If this timer interval expires before the router receives another group membership report, the router determines that the group has no members left on the network.

IGMPv3 supports an extended report format you can use to report multiple groups and source lists in a single report.

Leave Group Membership Messages

When a host leaves a group, it sends a leave group membership message to multicast routers on the network. A host generally addresses leave group membership messages to the all-routers group address (224.0.0.2).

Platform Considerations

For information about modules that support IGMP on the ERX-7xx models, ERX-14xx models, and the ERX-310 router:

- See *ERX Module Guide, Table 1, Module Combinations* for detailed module specifications.
- See *ERX Module Guide, Appendix A, Module Protocol Support* for information about the modules that support IGMP.

For information about modules that support IGMP on the E120 router and the E320 router:

- See *E120 and E320 Module Guide, Table 1, Modules and IOAs* for detailed module specifications.
- See *E120 and E320 Module Guide, Appendix A, IOA Protocol Support* for information about the modules that support IGMP.

References

For more information about IGMP, see the following resources:

- IGMP-based Multicast Forwarding ("IGMP Proxying")—draft-ietf-magma-igmp-proxy-00.txt (May 2002 expiration)
- RFC 2236—Internet Group Management Protocol, Version 2 (November 1997)
- RFC 2933—Internet Group Management Protocol MIB (October 2000)
- RFC 3292—General Switch Management Protocol (GSMP) V3 (June 2002)
- RFC 3376—Internet Group Management Protocol (October 2002)
- GSMP extensions for layer2 control (L2C) Topology Discovery and Line Configuration—draft-wadhwa-gsmp-l2control-configuration-00.txt (July 2006 expiration)

Before You Begin

You can configure IGMP on IPv4 multicast interfaces.

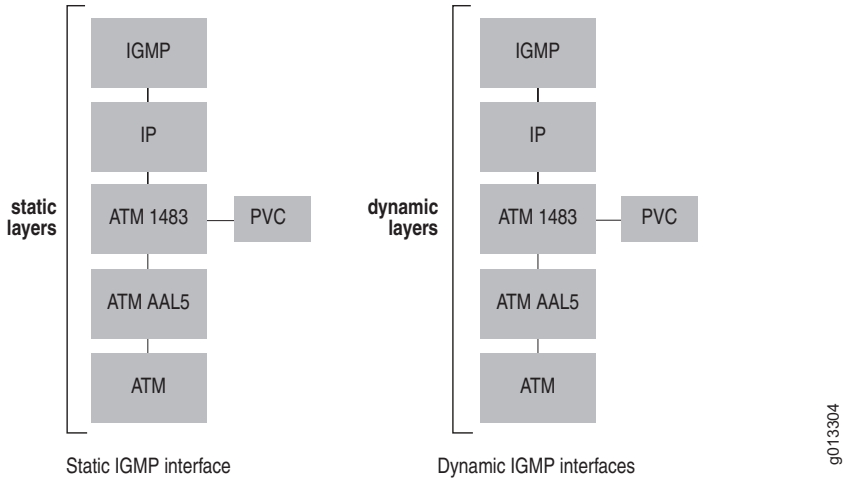
For information about IPv4 multicasting, see *Chapter 1, Configuring IPv4 Multicast*. For information about configuring IP interfaces, see *JUNOSe IP, IPv6, and IGP Configuration Guide, Chapter 1, Configuring IP*. For information about configuring IPv6 interfaces, see *JUNOSe IP, IPv6, and IGP Configuration Guide, Chapter 2, Configuring IPv6*.

For information about configuring multicast on IPv6 interfaces, see *Chapter 5, Configuring IPv6 Multicast*.

Configuring Static and Dynamic IGMP Interfaces

The router supports *static* and *dynamic* IGMP interfaces. Unlike static interfaces, dynamic interfaces are not restored when you reboot the router. For some protocols, dynamic layers can build on static layers in an interface; however, in a dynamic IGMP interface, all the layers are dynamic. See Figure 5 for examples of static and dynamic IGMP interfaces.

Figure 5: Static and Dynamic IGMP Interfaces



You configure static IGMP interfaces by using software such as the CLI or an SNMP application; you configure dynamic IGMP interfaces by using a profile. A profile constitutes a set of attributes for an interface; a profile for dynamic IGMP interfaces contains attributes for configuring all the layers in the interface.

You define a profile by using the same CLI commands that you use to configure a static IGMP interface; however, the mode in which you use the commands differs. Use the commands in Interface Configuration mode to configure a static IGMP interface and in Profile Configuration mode to define a profile.

When you have defined a profile, you can apply it to an interface or a group of interfaces. Profiles provide an efficient method of creating and managing large numbers of dynamic interfaces. For detailed information about creating and assigning profiles, see *JUNOSe Link Layer Configuration Guide, Chapter 12, Configuring Dynamic Interfaces*. When you create a profile for dynamic IGMP interfaces, specify attributes for configuring all layers in the interface.

You use the following IGMP commands to configure a static IGMP interface. You also use these commands to define the attributes for the IGMP layer when you create a profile for dynamic IGMP interfaces:

Table 6: IGMP Commands

ip igmp	ip igmp query-max-response-time
ip igmp access-group	ip igmp robustness
ip igmp access-source-group	ip igmp ssm-map enable
ip igmp apply-oif-map	ip igmp ssm-map static

Table 6: IGMP Commands (continued)

ip igmp explicit-tracking	**ip igmp query-interval**
ip igmp group limit	**ip igmp static-exclude**
ip igmp immediate-leave	**ip igmp static-group**
ip igmp last-member-query-interval	**ip igmp static-include**
ip igmp promiscuous	**ip igmp version**
ip igmp querier	**multicast group port limit**
ip igmp querier-timeout	

The following sections describe the tasks associated with these and other **ip igmp** commands.

You can also use various IGMP-specific RADIUS attributes in RADIUS Access-Accept messages as an alternative method of configuring certain values. See *JUNOSe Broadband Access Configuration Guide, Chapter 3, Configuring RADIUS Attributes* for additional information.

Enabling IGMP on an Interface

You must start IGMP on each interface that you want to use the protocol. You can configure IGMP and either PIM or DVMRP on the same interface. If you configure only IGMP on an interface, IGMP owns that interface. If you configure IGMP and either PIM or DVMRP on an interface, PIM or DVMRP owns the interface.

By enabling IGMP, the router processes incoming multicast packets and creates an entry in the multicast routing table. If neither PIM nor DVMRP own the interface (for example, when only IGMP is configured), then the packets are locally routed to other interfaces on the router. PIM or DVMRP must be configured on the interface for packets to be sent to other routers.

For networks that use only IGMPv1, you can configure an interface to operate in IGMPv1 mode. However, IGMPv2 and IGMPv3 interfaces support IGMPv1 hosts. In an IGMPv1 network, you must configure one interface to act as a querier. In an IGMPv2 or IGMPv3 network, the querier is the router with the lowest IP address.

To start IGMP, complete the following steps:

1. Enable IGMP on the interface (IGMPv2 is the default version).
2. (IGMPv1 or IGMPv3) Specify the IGMP version for the interface.
3. (IGMPv1 only) Specify that the interface act as the querier for the network.

ip igmp

- Use to enable IGMP on an interface and to set the IGMP version to IGMPv2. Use the **ip igmp version** command to specify a different IGMP version.
- Example

  host1:boston(config-if)#**ip igmp**

- Use the **no** version to disable IGMP on an interface.

ip igmp querier

- Use to specify that this IGMPv1 interface acts as a querier.

NOTE: This command is valid only for interfaces on which you configured IGMPv1.

- By default, IGMPv1 interfaces act as queriers.
- Example

  host1:boston(config-if)#**ip igmp querier**

- Use the **no** version to cause the interface to not act as a querier.

ip igmp version

- Use to set the IGMP version (1, 2, or 3) for the interface or specify a passive interface with only multicast-data-forwarding capability (passive).
- Example

  host1:boston(config-if)#**ip igmp version 1**

- Use the **no** version to set the version to the default, IGMPv2.

Configuring IGMP Settings for an Interface

When you start IGMP on an interface, it operates with the default settings. You can, however, modify:

- The method that the router uses to remove hosts from multicast groups (IGMPv2 and IGMPv3 interfaces only)
- The query time interval for the querier sends group membership messages
- The time that a non-querier waits for queries from the current querier before sending query messages to assume responsibility of querier
- The time that a new querier waits before sending query messages after it assumes responsibility from another querier
- The time that a host can take to reply to a query (maximum response time)
- The number of times that the router sends each IGMP message from this interface

ip igmp immediate-leave

- Use to specify that, when the router receives a leave group membership message from a host associated with this interface, the router immediately removes that host from the multicast group.

CAUTION: Issue this command only on IGMPv2 and IGMPv3 interfaces to which one IGMP host is connected. If more than one IGMP host is connected to a LAN through the same interface, and one host sends a leave group message, the router removes all hosts on the interface from the multicast group. The router loses contact with the hosts that must remain in the multicast group until they send join requests in response to the router's next general group membership query.

- Use the IGMP-Immediate-Leave RADIUS attribute (VSA 26-97) in RADIUS Access-Accept messages as an alternative method of configuring this value. The RADIUS setting takes precedence over a CLI setting.
- Example

  host1:boston(config-if)#**ip igmp immediate-leave**

- Use the **no** version to restore the default behavior, in which the router removes a host from a multicast group if that host does not return a group membership report within a certain length of time after receiving a group membership query from the router.

ip igmp last-member-query-interval

- Use to specify the last-member-query-interval value, in the range 1–255 tenths of a second. When the router receives an IGMPv2 leave message or an IGMPv3 state change report, it sends out a query and expects a response within the time specified by this value.
- Using a lower value enables members to leave groups more quickly.
- Example

  host1:boston(config-if)#**ip igmp last-member-query-interval 90**

- Use the **no** version to restore the default, 10-tenths of a second (1 second).

ip igmp querier-timeout

- Use to set the time, in the range 1–400 seconds, that the interface waits for queries from the current querier before sending query messages to assume responsibility of querier.
- Example

  host1:boston(config-if)#**ip igmp querier-timeout 200**

- Use the **no** version to set the time to the default, twice the query interval.

ip igmp query-interval

- Use to specify how often, in the range 1–300 seconds, the interface sends group membership queries.
- Use the IGMP-Query-Interval RADIUS attribute (VSA 26-95) in RADIUS Access-Accept messages as an alternative method of configuring this value. The RADIUS setting takes precedence over a CLI setting.
- Example

  host1:boston(config-if)#**ip igmp query-interval 100**

- Use the **no** version to set the polling interval to the default, 125 seconds.

ip igmp query-max-response-time

- Use to specify the time in tenths of a second in which the host must respond to a group membership query. The possible period ranges are as follows:
  - IGMPv2: 1–255 tenths of a second
  - IGMPv3: 1–31744 tenths of a second
- IGMPv2 and IGMPv3 include this value in IGMP query messages sent out on the interface.
- You cannot set this value on interfaces running IGMPv1.
- Using a lower value enables members to join and leave groups more quickly.
- Use the IGMP-Max-Resp-Time RADIUS attribute (VSA 26-96) in RADIUS Access-Accept messages as an alternative method of configuring this value. The RADIUS setting takes precedence over a CLI setting.
- Example

  host1:boston(config-if)#**ip igmp query-max-response-time 120**

- Use the **no** version to restore the default, 100 tenths of a second (10 seconds).

ip igmp robustness

- Use to specify the number of times that the router sends each IGMP message from this interface.
- Use a higher value to ensure high reliability from IGMP.
- Specify a number in the range 1–4.
- Example

  host1:boston(config-if)#**ip igmp robustness 2**

- Use the **no** version to restore the default, 3.

Specifying Multicast Groups

You can use a standard-format or extended-format IP access list to specify the multicast groups that a host can join.

ip igmp access-group

- Use to restrict hosts on this subnetwork to join only multicast groups that appear on the specified IP access list.
- When this feature is configured, the access list is queried whenever the router receives an IGMPv2 report requesting membership of a group, and IGMPv3 ChangeToInclude or IsExclude reports. The request is rejected if the access list query fails.
- The **ip igmp access-group** command accepts standard or extended-format access lists. Because the extended format enables you to specify both the source address and the destination group address, the source address must be set to any. For example, **access-list test permit ip host 224.128.64.32 any**.
- Note that in the access list specified when you issue this command, the group is specified before the source.
- Example

  host1:boston(config-if)#**ip igmp access-group boston-list**

- Use the **no** version to dissociate the interface from an access list and to enable hosts on the interface to join any multicast group.

ip igmp access-source-group

- Use to restrict hosts on this subnetwork to membership in those (S,G) pairs (also known as *channels*) included on the specified IP access list.
- When this feature is configured, both source and group addresses query the associated access list whenever the router receives an IGMPv3 report requesting membership of the (S,G) pairs (that is, the router receives an IGMPv3 ChangeToInclude, IsInclude, or AllowNewSource group report). The request is rejected if the access list query fails.
- The **ip igmp access-source-group** command accepts standard or extended-format access lists. The extended format enables you to specify both the source address and the destination group address; for example, **access-list test permit ip host 10.1.1.1 host 224.128.64.32**. Typically, you use the extended-format access list. If you instead use the standard-format access list, you explicitly specify the source address to create the access list, but the group address is implicitly assumed to be **any**,
- Note that in the access list specified when you issue this command, the source is specified before the group.
- Example

  host1:boston(config-if)#**ip igmp access-source-group dallas-list**

- Use the **no** version to remove any access list restriction.

Assigning a Multicast Group to an Interface

You can assign an interface to send and receive all traffic for a particular multicast group. This feature enables you to control the IGMP traffic and to test the behavior of multicast protocols in the network.

ip igmp static-group

- Use to send and receive all traffic for a multicast group from a specific interface.
- The interface sets no timers for this group.
- Example

  host1:boston(config-if)#**ip igmp static-group 225.1.2.3**

- Use the **no** version to stop the interface from sending all traffic for the group.

Configuring Group Outgoing Interface Mapping

You can configure an IGMP interface to use a different outgoing interface (OIF) for multicast-date-forwarding by applying an OIF map. When you configure an OIF map on an IGMP interface, the map is applied to all IGMP membership requests that the interface receives.

To configure OIF mapping on an interface:

1. Create an OIF map using the **ip igmp oif-map** command at the global level.
2. Apply the OIF map to an interface using the **ip igmp apply-oif-map** command.

To properly configure an interface used in the OIF map for multicast-data-forwarding capability, you must configure the interface version as passive with the **ip igmp version** command. You can either specify a passive interface as the OIF or specify the OIF as *self* (to use the IGMP interface as the OIF) in the **ip igmp oif-map** command.

ip igmp apply-oif-map

- Use to apply the specified outgoing interface (OIF) map to the current interface.
- Example

  host1(config-subif)#**ip igmp apply-oif-map OIFMAP**

- Use the **no** version to remove the outgoing interface map association from the interface.

ip igmp oif-map

- Use to create an OIF map.
- Example

  host1(config)#**ip igmp oif-map OIFMAP atm 3/0.1 232.0.0.1/32 51.0.0.1/32**
  host1(config)#**ip igmp oif-map OIFMAP atm 3/0.2 232.0.0.1/32 51.0.0.2/32**
  host1(config)#**ip igmp oif-map OIFMAP atm 3/0.3 233.0.0.1/32**
  host1(config)#**ip igmp oif-map OIFMAP atm 3/0.4 233.0.0.0/24 51.0.0.1/32**
  host1(config)#**ip igmp oif-map OIFMAP atm 3/0.5 233.0.0.0/24 51.0.0.2/32**
  host1(config)#**ip igmp oif-map OIFMAP self 0.0.0.0/0 51.0.0.0/24**

- Use the **no** version to remove an outgoing interface map attribute.

ip igmp version

- Use to set the IGMP version (1, 2, or 3) for the interface or specify a passive interface with only multicast-data-forwarding capability (passive).
- Example

  host1:dallas(config-if)#**ip igmp version passive**

- Use the **no** version to set the version to the default, IGMPv2.

Configuring Access Node Control Protocol for IGMP

By using ANCP, IGMP is no longer terminated or proxied at the access node. Instead, IGMP passes through the access node transparently. B-RAS terminates both the data PVC and IGMP. After possible user permission verification, B-RAS may instruct the access node, by using GSMP, to establish a multicast branch for the subscriber port.

L2C works with a special IGMP session to collect OIF mapping events in a scalable manner. For additional information about configuring L2C for IGMP, see *JUNOSe IP Services Configuration Guide, Chapter 8, Configuring ANCP*.

For additional information about OIF mapping, see *Configuring Group Outgoing Interface Mapping* on page 53.

Configuring SSM Mapping

Source-specific multicast (SSM) mapping enables the router to determine one or more source addresses for group G. The mapping effectively translates IGMPv1 or IGMPv2 membership reports to an IGMPv3 report, enabling the router to continue as if it had initially received an IGMPv3 report. After the router is joined to these groups, it sends out PIM join messages and continues to enable joining from these groups, as long as it continues to receive IGMPv1 and IGMPv2 membership reports and no change occurs to the SSM mapping for the group.

When you statically configure SSM mapping, the router can discover source addresses from a statically configured table.

The following conditions apply when you configure SSM mapping:

- When SSM mapping is enabled, and either you have not configured a static SSM map or the router cannot find any matching access lists, the router continues to accept (*,G) groups. The PIM SSM range must deny any unacceptable SSM group addresses.
- When you issue the **no ip igmp ssm-map enable** command, the router removes all SSM map (S,G) states and establishes a (*,G) state.
- You can enter multiple **ssm-map static** commands for different access lists. Also, you can enter multiple **ssm-map static** commands for the same access list, as long as the access list uses different source addresses.
- SSM maps do not process statically configured groups.

ip igmp ssm-map enable

- Use to enable SSM mapping on the router. SSM mapping statically assigns sources to IGMPv1 and IGMPv2 groups. You must use SSM mapping for IGMPv1 and IGMPv2 hosts to interoperate with PIM SSM. SSM mapping enables the router to use a statically configured list to translate (*,G) memberships to (S,G) memberships.
- Example

  host1:boston(config)#**ip igmp ssm-map enable**

- Use the **no** version to disable SSM mapping on the router.

ip igmp ssm-map static

- Use to specify an access list and source address for use in SSM mapping. SSM mapping statically assigns sources to IGMPv1 and IGMPv2 groups. You must use SSM mapping for IGMPv1 and IGMPv2 hosts to interoperate with PIM SSM. SSM mapping enables the router to use a statically configured list to translate (*,G) memberships to (S,G) memberships.
- Example

  host1:boston(config)#**ip igmp ssm-map static boston-list 51.0.0.1**

- Use the **no** version to remove the SSM map association.

Limiting the Number of Accepted IGMP Groups

By default, there is no limit on the number of IGMP groups that an IGMP interface can accept. However, you can manage multicast traffic on the router by restricting the number of IGMP groups accepted by:

- A specific port on an I/O module
- A specific IGMP interface

If you set limits for both a port and interfaces on that port, the router uses the lower of the two limits when determining how many IGMP groups an interface can accept. For example, if you set a limit of 10 groups for the port and 15 groups for each interface, only 10 groups can be accepted among the interfaces.

However, if you set a limit for a port and that limit is lower than the number of groups currently accepted by the interfaces on that port, the router does not dissociate the groups from the interfaces. The router enforces the new limit on the port when the number of groups associated with the interfaces falls to that limit. For example, if the interfaces on the port have accepted a total of 15 groups, and you set a limit of 10 groups on the port, the router does not disconnect any of the groups and prevents the interfaces from accepting any more groups. Over time, some groups leave the interfaces and, eventually, a maximum of ten groups remain connected.

ip igmp group limit

- Use to limit the number of IGMP groups that an interface can accept.

NOTE: This command is deprecated and might be removed completely in a future release. The function provided by this command has been replaced by the updated by limiting bandwidth of multicast streams using the **ip multicast admission-bandwidth-limit** command.

- Example

  host1:boston(config-if)#**ip igmp group limit 5**

- Use the **no** version to restore the default behavior, in which there is no limit on the number of IGMP groups that an interface can accept.

multicast group port limit

- Use to limit the number of IGMP groups that a port can accept.

NOTE: This command is deprecated and might be removed completely in a future release. The function provided by this command has been replaced by the updated by limiting bandwidth of multicast streams using the **mroute port admission-bandwidth-limit** command.

- Specify the identifier for the port in *slot/port* format (ERX routers) or in *slot/adapter/port* format (E320 router).
  - *slot*—Number of the chassis slot in the range 0–6 (ERX-7xx models), 0–13 (ERX-14xx models), or 0–16 (E320)
  - *adapter*—Adapter number on the E320 IOA module
  - *port*—Port number on the I/O or IOA module
- Specify the maximum number of IGMP groups that interfaces can accept.
- Example 1—ERX models

  host1(config)#**multicast group port 3/0 limit 5**

- Example 2—E320 router

  host1(config)#**multicast group port 3/1/0 limit 5**

- Use the **no** version to restore the default behavior, in which there is no limit on the number of IGMP groups that a port can accept.

Including and Excluding Traffic

IGMPv3 extends IGMPv2 functionality with the ability to include or exclude specific multicast traffic sources. That is, with IGMPv3, hosts signal (S,G) pairs to be included or excluded.

For hosts that cannot signal group membership dynamically, you can use the **ip igmp static-include** or **ip igmp static-exclude** command to statically include or exclude multicast traffic, respectively.

IGMPv3 is the industry-designated standard protocol for hosts to signal channel subscriptions in SSM. For additional information about SSM, see *PIM Source-Specific Multicast* on page 83.

ip igmp static-exclude

- Use to statically exclude the IGMP (S,G) membership for a host that is not capable of dynamically signaling group membership.
- Example

  host1:boston(config-if)#**ip igmp static-exclude 10.1.1.5 225.1.2.3**

- Use the **no** version to remove the static designation.

ip igmp static-include

- Use to statically include the IGMP (S,G) membership for a host that is not capable of dynamically signaling group membership.
- Example

  host1:boston(config-if)#**ip igmp static-include 10.1.1.1 225.1.2.3**

- Use the **no** version to remove the static designation.

Configuring Explicit Host Tracking

Explicit host tracking enables the router to explicitly track each individual host that is joined to a group or channel on a particular multi-access network.

Explicit host tracking provides the following:

- Minimal leave latency when a host leaves a multicast group or channel. When the router receives a leave message for a group or channel on an interface, it accesses a list of hosts and immediately stops forwarding traffic if the sender is the last host to request traffic for that group or channel. The leave latency is bound only by the packet transmission latencies in the multi-access network and the processing time in the router.
- Ability to change channels quickly in networks where bandwidth is constrained between a multicast-enabled router and hosts.
- Ability to determine what multicast hosts are joined to particular multicast groups or channels, which is useful for accounting purposes.
- Reduction of control message traffic on the network because, when it receives a leave message, the router no longer needs to send out IGMP queries to verify membership. As a result, interested hosts also do not need to respond to these queries with reports.
- Tracking based on the IGMP reports for hosts in both include and exclude modes for every multicast group or channel on an interface.

When the router is configured for explicit host tracking and starts immediate leave using the host information collected, every leave message received for a group or channel is treated as follows:

- The router checks the number of hosts that receive traffic from the group or channel.
- If the host sending the leave message is the only host, it starts immediate leave for that group or channel on that interface. The router removes the interface from the multicast group or channel immediately, without sending out a group or group-source-specific query and waiting for the last member query interval.
- If the host sending the leave message is not the only host receiving traffic for that group or channel, the router removes the host from the list of hosts on that interface, but keeps the interface in the outgoing interface list for the multicast group or channel. No group or group-source-specific queries are sent.

If one or more hosts that support only IGMP V1 are present on a network, the leave latencies for the multicast groups to which those hosts are joined revert to the IGMP V1 leave latency. This affects only the multicast groups to which these legacy hosts are actually joined at any point in time.

You cannot configure explicit host tracking on passive IGMP interfaces or on IGMP V1 interfaces. When you enable IGMP V2 or V3 on an interface, explicit host tracking is not enabled by default.

When you enable explicit host tracking on an interface that has a membership state, the router does not immediately start performing immediate leave. For a maximum of group membership interval seconds, the router only performs host tracking. Any leave messages that the router receives during this period receive normal leave processing. Any leave messages received after this interval has elapsed receive immediate leave processing, when appropriate.

When explicit host tracking has been enabled on an IGMP V3 interface, even if a group has to downgrade to IGMP V2 due to the presence of an IGMP V2 host, explicit host tracking continues for that group. To avoid this, you can use the **disable-if-igmp-v2-detected** keyword. If you select this option, the router turns off explicit host tracking for the group when IGMP V2 host reports are received for the group on that interface. This option does not have any significance on an interface configured for IGMP V2 and is ignored if provided. Because IGMP V1 does not support leave messages, explicit host tracking is turned off for a group that downgrades to IGMP V1 due to the presence of IGMP V1 hosts.

Explicit host tracking cannot be enabled on an interface that has immediate-leave configured and vice versa. Any attempt to configure immediate-leave on an interface that has explicit host tracking enabled or to configure explicit host tracking on an interface that has immediate-leave enabled is rejected and an error message logged on the screen.

The following example enables IGMP V3 explicit host tracking on interface 3/0.101 with the default configuration where the router continues to perform explicit host tracking for IGMP V2 groups. To override this default configuration, you must use the **ip igmp explicit-tracking disable-if-igmp-v2-detected** command.

```
interface 3/0.101
ip igmp version 3
ip igmp explicit-tracking
end
```

ip igmp explicit-tracking

- Use to set explicit host tracking for IP IGMP interfaces.
- To disable explicit host tracking if IGMP V2 hosts are detected, use the **disable-if-igmp-v2-detected** keyword.
- Example

  host1(config)#**ip igmp explicit-tracking**

- Use the **no** version to disable explicit host tracking on the interface. Use the **no** version with the **disable-if-igmp-v2-detected** keyword to revert to the default explicit host tracking behavior.

Accepting IGMP Reports from Remote Subnetworks

By default, IGMP interfaces accept IGMP reports only from associated subnetworks. You can configure the router to accept IGMP reports from subnetworks that are not associated with its interfaces. The **igmp promiscuous** command in Router Configuration mode specifies whether interfaces on the router can accept IGMP reports from indirectly connected subnets. To override this global setting on a particular interface, use the **ip igmp promiscuous** command in Interface Configuration mode.

Example In the following example, the router is configured to accept IGMP reports from indirectly connected subnets on all interfaces. The interface on port 0 of the line module in slot 4 is then configured to accept IGMP reports only from directly connected subnets.

```
host1(config)#virtual-router boston
host1:boston(config)#router igmp
host1:boston(config-router)#igmp promiscuous
host1:boston(config-router)#exit
host1:boston(config)#interface serial 4/0
host1:boston(config-if)#ip igmp promiscuous off
```

igmp promiscuous

- Use to enable all IGMP interfaces on the router to accept IGMP reports from hosts on any subnetwork.
- Example

  ```
  host1:boston(config-router)#igmp promiscuous
  ```

- Use the **no** version to enable IGMP interfaces on the router to accept IGMP reports only from hosts on their associated subnetworks.

ip igmp promiscuous

- Use to specify whether the interface accepts IGMP reports from hosts on any subnetwork.
  - Use the **on** keyword to enable the interface to accept IGMP reports from hosts on any subnetwork.
  - Use the **off** keyword to enable the interface to accept IGMP reports only from hosts on subnetworks associated with this interface.
- Example

  ```
  host1:boston(config-if)#ip igmp promiscuous on
  ```

- Use the **no** version to configure an IGMP interface to use the Router Configuration mode setting to determine the subnetworks from which it can accept IGMP reports.

Disabling and Removing IGMP

You can disable and reenable IGMP on the VR. You can also remove IGMP from the VR and recreate it on the VR.

igmp disable

- Use to disable IGMP on a VR.
- Example

  host1(config)#**virtual-router boston**
  host1:boston(config)#**router igmp**
  host1:boston(config-router)#**igmp disable**

- Use the **no** version to enable IGMP on a VR.

router igmp

- Use to create and enable IGMP on a VR or to access IGMP Router Configuration mode.
- Example

  host1(config)#**virtual-router boston**
  host1:boston(config)#**router igmp**

- Use the **no** version to remove IGMP and the IGMP proxy from the VR.

Monitoring IGMP

You can establish a reference point for IGMP statistics by setting the statistics counters to zero.

To display IGMP parameters, use the **show** commands described in this section.

NOTE: The E120 router and E320 router output for **monitor** and **show** commands is identical to output from other E-series routers, except that the E120 and E320 router output also includes information about the adapter identifier in the interface specifier (*slot/adapter/port*).

baseline ip igmp

- Use to set the counters for IGMP statistics to zero, to establish a baseline.
- Example

  (host1)#**baseline ip igmp**

- There is no **no** version.

show ip igmp

- Use to display IGMP information for a VR.
- Field descriptions
  - Routing Process—Routing process for this VR (IGMP)
  - Administrative state—Status of IGMP in the software: enabled or disabled
  - Operational state—Status of IGMP on the VR: enabled or disabled
  - Total interfaces—Number of interfaces on which you started IGMP
  - enabled—Number of interfaces on which IGMP is enabled
  - disabled—Number of interfaces on which IGMP is disabled
  - learnt groups—Number of multicast groups that the VR has discovered
  - IGMP graceful restart duration—Restart interval in seconds
  - IGMP Statistics Rcvd—Statistics for IGMP messages received
    - total—Total number of IGMP messages received
    - checksum errors—Number of IGMP messages received with checksum errors
    - unknown types—Number of IGMP messages received that are not group membership queries, group membership reports, or leave group membership messages
    - queries—Number of group membership queries
    - reports—Number of group membership reports
    - leaves—Number of leave group membership messages
  - IGMP Statistics Sent—Statistics for IGMP messages sent
    - Total number of group membership queries sent
- Example

```
host1:boston#show ip igmp
Routing Process IGMP, Administrative state enabled, Operational state
enabled
    2 total interfaces, 2 enabled, 0 disabled
    0 enabled interfaces performing graceful restart
    2 learnt groups
IGMP Statistics:
    Rvcd: 1 total, 0 checksum errors, 0 unknown types
          0 queries, 1 reports, 0 leaves
    Sent: 11 total
```

show ip igmp groups

- Use to display statically joined and directly connected groups learned through IGMP.
- Field descriptions
  - Grp Address—Address of the multicast group
  - Interface—Interface that discovered the multicast group
  - oif-map—Name of the OIF map and the mapped OIF interface, when a group or source has been mapped to an OIF

- State—IGMP version on the interface
- ExpTim—Time, in seconds, at which the router stops polling for more members of this group
- oldHTo—Time at which the router stops polling for more IGMPv1 members of a group. If this value is 0, the interface has received no IGMPv1 reports for the group.
- Included Sources—Sources included in the multicast group
- Excluded Sources—Sources excluded from the multicast group
- Counts—Number of source-group mappings by version and state

- Example 1—Without OIF mapping

```
host1:boston#show ip igmp groups
Grp Address     Interface       State       Reporter        ExpTim oldHTo
--------------- --------------- ----------- --------------- ------ ------
228.1.1.1       FastEthernet1/1 Version3    17.0.0.2            44      0
228.1.1.2       FastEthernet1/1 Version3    17.0.0.2            50      0
228.1.1.3       FastEthernet1/1 Version3    17.0.0.2            48      0
230.1.1.1       FastEthernet1/1 Version3    17.0.0.2            44      0
  Included Sources:
    51.0.0.1                                                    44
    51.0.0.2                                                    44
    51.0.0.3                                                    44
231.1.1.1       FastEthernet1/1 Version3    17.0.0.2            44      0
  Excluded Sources:
    51.0.0.1                                                     0
    51.0.0.2                                                     0
    51.0.0.3                                                     0

Counts: 5 version-3, 0 version-2, 0 version-1, 0 check state, 0 disabled
       (5 total)
        0 excluded
        Source-groups: 3 included, 3 excluded
```

- Example 2—With OIF mapping

```
host1:boston#show ip igmp groups
Grp Address     Interface       State       Reporter        ExpTim oldHTo
--------------- --------------- ----------- --------------- ------ ------
232.1.1.1       ATM5/0.12       Version3    1.1.1.2            371      0
                    oif-map OIFMAP ATM5/0.121
232.1.1.1       ATM5/0.13       Version3    1.1.1.3            375      0
                    oif-map OIFMAP ATM5/0.121
232.1.1.2       ATM5/0.12       Version3    1.1.1.2            373      0
  Included Sources:
    10.1.1.2        oif-map OIFMAP self                        373
    10.1.1.10       oif-map OIFMAP ATM5/0.120                  373
    10.1.1.11       oif-map OIFMAP ATM5/0.121                  373
232.1.1.2       ATM5/0.13       Version3    1.1.1.3            375      0
  Included Sources:
    10.1.1.2        oif-map OIFMAP self                        375
    10.1.1.10       oif-map OIFMAP ATM5/0.120                  375
    10.1.1.11       oif-map OIFMAP ATM5/0.121                  375

Counts: 4 version-3, 0 version-2, 0 version-1, 0 check state, 0 disabled
       (4 total)
        0 excluded
        Source-groups: 6 included, 0 excluded
```

show ip igmp interface

- Use to display IGMP information for interfaces on which you enabled IGMP.
- Specify the **brief** keyword to see a summary of the information.
- Specify the **count** keyword to see the number of IGMP interfaces.
- Specify the **group** address keyword to see information for interfaces that belong to that group.
- Field descriptions
  - Interface—Type of interface and interface specifier. For details about interface types and specifiers, see *JUNOSe Command Reference Guide, About This Guide*.
  - address—IP address of the interface
  - Administrative state—Status of the interface in the software: enabled or disabled
  - Operational state—Physical status of the interface: enabled or disabled
  - Version—IGMP version
  - State—Function of the interface: querier or nonquerier
  - Query Interval—Time interval in seconds at which this interface sends query messages
  - Other querier present interval—Time in seconds that the interface waits before declaring itself as the querier
  - Maximum response time—Time interval, in tenths of a second, during which this interface waits for a host to respond
  - Last member query interval—Time, in tenths of a second, that this interface waits before sending a new query to a host that sends a group leave message
  - Robustness—Number of times this interface sends IGMP messages
  - Information about whether the interface accepts IGMP reports from hosts on any subnetwork
    - Interface defaults to global promiscuous mode—Interface uses the setting of the **igmp promiscuous** command to determine whether it accepts IGMP reports from hosts on any subnetwork
  - Information about standard IP access lists configured with the **ip igmp access-group** command
    - Inbound access group—Access list specified
    - No inbound access group—No access list specified
  - Information about IP access lists configured with the **ip igmp access-source-group** command
    - Inbound access source-group—Access list specified
    - No inbound access source-group—No access list specified

- Information about OIF maps configured with the **ip igmp apply-oif-map** command
  - Inbound apply-oif-map—Map name specified
  - No inbound apply-oif-map—No map name specified
- Immediate Leave—Setting of the **ip igmp immediate-leave** command: enabled or disabled
- Explicit Host Tracking—Setting of the **ip igmp explicit-tracking** command: enabled or disabled
- Max-Group limit—Number of IGMP groups that the interface can accept, as configured with the **ip igmp group limit** command
- Admission-Bandwidth limit—Value of the admission-bandwidth limit set for an interface that accepts IGMP groups, or No Limit
- Group Count—Number of IGMP groups that the interface has accepted
- IOA packet replication—Hardware multicast packet replication interface to which egress multicast packets on this interface are redirected
- Interface statistics Rcvd—Information about IGMP messages received on this interface
  - reports—Number of group membership reports received
  - leaves—Number of group leave messages received
  - wrong version queries—Number of group membership queries received from devices running a different version of IGMP
- Interface statistics Sent—Number of IGMP messages this interface has sent
- Interface statistics Groups learned—Number of groups this interface has discovered
- Counts—Breakdown of IGMP interfaces
  - down—Number of interfaces down
  - init state—Number of interfaces in the initialization state
  - querier—Number of querier interfaces
  - non-querier—Number of non-querier interfaces
  - Total—Total number of IGMP interfaces

- Example 1

```
host1:boston#show ip igmp interface
Interface ATM2/1.15 address 15.0.0.2/255.255.255.0
Administrative state enabled, Operational state enabled
Interface parameters:
    Version 2
    State Querier
    Query Interval 125 secs, 53 secs before the next query
    Other querier present interval 250 secs
    Maximum response time 100 (in 10ths of a second)
    Last member query interval 10 (in 10ths of a second)
    Robustness 3
    Interface defaults to global promiscuous mode
    No inbound access group
    No inbound access source-group
    No inbound apply-oif-map
```

```
    Immediate Leave: disabled
    Explicit Host Tracking: enabled
    Max-Group limit: No Limit
    Admission-Bandwidth limit: No Limit
    Group Count: 1
Interface statistics:
    Rcvd: 0 reports, 0 leaves, 0 wrong version queries
    Sent: 1 queries
    Groups learned: 1

Counts: 0 down, 0 init state, 1 querier, 0 non-querier, 1 Total
Counts: 0 down, 0 init state, 1 querier, 0 non-querier, 1 Total
```

- Example 2

```
host1#show ip igmp interface gigabitEthernet 3/0.0
Interface GigabitEthernet3/0.0 address 10.1.1.1/255.255.255.0
Administrative state enabled, Operational state disabled
Interface parameters:
    Version 2
    State Down
    Query Interval 125 secs
    Other querier present interval 250 secs
    Maximum response time 100 (in 10ths of a second)
    Last member query interval 10 (in 10ths of a second)
    Robustness 3
    Interface defaults to global promiscuous mode
    No inbound access group
    No inbound access source-group
    No inbound apply-oif-map
    Immediate Leave: disabled
    Explicit Host Tracking: enabled
    Max-Group limit: No Limit
    Admission-Bandwidth limit: No Limit
    Group Count: 0
    IOA packet replication gigabitEthernet 3/8.1
Interface statistics:
    Rcvd: 0 reports, 0 leaves, 0 wrong version queries
    Sent: 0 queries
    Groups learned: 0
```

show ip igmp interface brief

- Use to display a summary of IGMP information for interfaces on which you enabled IGMP.
- Field descriptions
  - Interface—Type of interface and interface specifier. For details about interface types and specifiers, see *JUNOSe Command Reference Guide, About This Guide*.
  - Intf Address—IP address of the interface
  - Ver—IGMP version
  - State—Function of the interface: querier or nonquerier
  - Querier—IP address of the querier on the network to which this interface connects

- QTime—Time interval, in seconds, at which this interface sends query messages
- QPTime—Time in seconds that the interface waits before declaring itself as the querier
- Count—Total number of IGMP interfaces

- Example

```
host1:boston#show ip igmp interface brief
Interface        Intf Address      Ver  State    Querier         QTime  QPTime
--------------   ----------        ---  ------   -------------   -----  ---
fastEthernet0/0  192.168.1.250/24  2    Querier  192.168.1.250   28     0
atm3/0.2         21.1.1.1/8        2    Querier  21.1.1.1        26     0
Count: 2 interfaces
```

show ip igmp mapped-oif

- Use to display the current mappings to all mapped outgoing interfaces or to the specified mapped outgoing interface.
- Field descriptions
  - OIF—Outgoing interface used in an OIF map
  - Oper—Operation status of the outgoing interface
  - Group Address—Multicast group IP address associated with the OIF
  - Source Address—Source IP address associated with the OIF
  - Join I/F—IGMP interface associated with the OIF
  - Map Name—Name of the map associated to the OIF
  - Counts—Number of source-group mappings to OIFs
- Example

```
host1#show ip igmp mapped-oif
      OIF       Oper  Group Address  Source Address     Join I/F      Map Name
--------------- ---- --------------- --------------- --------------- ----------
ATM5/0.120      Up   232.1.1.2       10.1.1.10       ATM5/0.12       OIFMAP
                                                     ATM5/0.13       OIFMAP
ATM5/0.121      Up   232.1.1.1       *               ATM5/0.12       OIFMAP
                                                     ATM5/0.13       OIFMAP
                     232.1.1.2       10.1.1.11       ATM5/0.12       OIFMAP
                                                     ATM5/0.13       OIFMAP

Counts: 3 source-group mappings
```

show ip igmp membership

- Use to display IGMP membership information for multicast groups and (S, G) channels.
- Specify the **tracked** keyword to see interface information only for interfaces where explicit host tracking is enabled.

- Field descriptions
  - Group—Multicast group or (S, G) channel
  - Source—(S, G) entries that are forwarding traffic
  - Reporter—Hosts that requested including sources or have not requested excluding sources. If listed under a group, host that sent exclude reports for the group. If listed under a source, host that requested traffic from this source for the group. For any (S, G), if listed under a source, indicates hosts interested in the traffic for this (S, G).
  - ExpTim—Expiration time.
  - Flags
    - M—Uses Oifmap
    - S—SSM mapped
    - T—Tracked
    - 1, 2, 3—IGMP version that the group is in
  - Interface—Type of interface and interface specifier. For details about interface types and specifiers, see *JUNOSe Command Reference Guide, About This Guide*.
- Example

```
host1#show ip igmp membership
Flags: M - Uses Oifmap  S- SSM mapped  T - tracked
       1,2,3 - The version of IGMP the group is in
Reporter:
    <ip-address> - last reporter if the group is not explicitly tracked
    <n>/<m> - <n> reporters include mode, <m> reporters in exclude

Group           Source            Reporter         ExpTim    Flags     Interface
--------------  ---------------   -------------    ------    ------    ---------------
224.0.1.40          *             10.10.1.1        02:41     2S        FastEthernet2/1
224.0.1.50                        1/2              02:56     3MT       FastEthernet2/2
                                  11.10.0.21       02:56
                                  11.10.2.22       02:30
                  20.30.0.11
                                  11.10.0.23       02:48
                  20.30.0.12
                                  11.10.0.21       02:56
                  20.30.0.13
                                  11.10.0.21       02:56
                                  11.10.0.22       02:30
                                  11.10.0.23       02:48
224.0.1.60                        20.20.0.1        01:56     3         FastEthernet2/3
                  10.30.0.100                      02:45
                  10.30.0.101                      02:35
                  10.30.0.102                      02:15
                  10.30.0.104                      stop
224.0.1.70                        30.20.0.1        stop      3         FastEthernet2/4
                  40.30.0.100                      01:10
                  40.30.0.101                      01.24
239.0.1.80                        2/0              stop      3T        FastEthernet2/5
                  50.30.0.100
                                  10.10.0.10       02:48
```

```
                50.30.0.101
                                    10.10.0.20       02:56
                                    10.10.0.10       02:48
                50.30.0.102         10.10.0.20       02:56
235.0.1.90                          0/3              02:56     2T       FastEthernet2/6
                     *
                                    12.10.0.10       02:48
                                    12.10.0.20       02:56
                                    12.10.0.30       02:48
```

show ip igmp oif-map

- Use to display all outgoing interface (OIF) maps or the OIF map for the specified map name.
- Field descriptions
  - Map Name—Name of the map associated to the show output
  - Group Prefix—Multicast group IP prefix
  - Source Prefix—Source IP prefix
  - OIF—Outgoing interface associated with the group and source prefix
- Example

```
host1#show ip igmp oif-map

     Map Name         Group Prefix      Source Prefix           OIF
------------------ ------------------ ------------------ ------------------
OIFMAP             232.1.1.0/24       0.0.0.0/0          ATM5/0.121
                   232.1.1.0/24       10.1.1.2/32        self
                   232.1.1.0/24       10.1.1.10/32       ATM5/0.120
                   232.1.1.3/32       0.0.0.0/0          ATM5/0.130
                   232.1.1.4/32       0.0.0.0/0          ATM5/0.130
```

show ip igmp oif-mapping

- Use to display the mapped OIF that is assigned to a given map-name, group address, and source address.
- Field descriptions
  - OIF-MAP Name—Name of the map requested
  - Group Address—Multicast group IP address requested
  - Source Address—Source IP address requested
  - Mapped OIF—Interface associated with the OIF map
- Example

```
host1#show ip igmp oif-mapping OIFMAP 232.1.1.1 10.1.1.10

    OIF Mapping
    OIF-MAP Name   : OIFMAP
    Group Address  : 232.1.1.1
    Source Address : 10.1.1.10
    Mapped OIF     : ATM5/0.120
```

show ip igmp ssm-mapping

- Use to display the SSM mapping state and the source list mapping associated with a multicast group address.
- Field descriptions
  - SSM Mapping—Status of SSM mapping on the interface: Enabled or Disabled
  - Group Address—Multicast group address requested
  - Source List—List of sources mapped to the multicast group address
- Example

```
host1:boston#show ip igmp ssm-mapping 232.1.1.1

    SSM Mapping   : Enabled
    Group Address : 232.1.1.1
    Source List   : 172.1.1.1
                  : 172.1.1.2
```

show multicast group limit

- Use to display the number of IGMP groups that ports have accepted and, if configured, the maximum number of groups that ports can accept.
- A value of –1 indicates that no port group limit is configured.
- Only ports that have accepted IGMP groups and ports for which you have configured a limit for the number of IGMP groups appear in this display.
- Field descriptions
  - Port—Identifier of the port in *slot*/*port* format
    - *slot*—Number of the chassis slot in the range 0–6 (ERX-7xx models) and 0–13 (ERX-14xx models)
    - *port*—Port number on the I/O module
  - limit—Maximum number of IGMP groups that the port can accept. A value of –1 indicates that no limit has been specified.
  - count—Actual number of IGMP groups that the port has accepted
- Example

```
host1:boston#show multicast group limit
Port      limit count
--------- ----- -----
2/0           5     0
2/1          -1     1
```

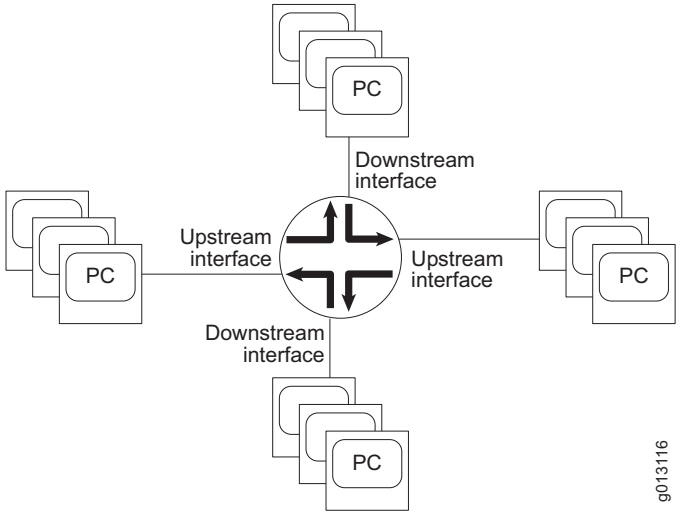
IGMP Proxy Overview

IGMP proxy enables the router to issue IGMP host messages on behalf of hosts that the router discovered through standard IGMP interfaces. The router acts as a *proxy* for its hosts. E-series routers support IGMP proxy versions 2 and 3.

Figure 6 shows a router in an IGMP proxy configuration. You enable IGMP proxy on one interface, which connects to a router closer to the root of the tree. This interface is the *upstream interface*. The router on the upstream interface is running IGMP.

You enable IGMP on the interfaces that connect the router to its hosts that are farther away from the root of the tree. These interfaces are known as *downstream interfaces*.

Figure 6: Upstream and Downstream Interfaces



As described in *IGMP Overview*, earlier in this chapter, hosts interact with the router through the exchange of IGMP messages. Similarly, when you configure IGMP proxy, the router interacts with the router on its upstream interface through the exchange of IGMP messages. However, when acting as the proxy, the router performs the host portion of the IGMP task on the upstream interface, as follows:

- When queried, sends group membership reports to the group.
- When one of its hosts joins a multicast address group to which none of its other hosts belong, sends unsolicited group membership reports to that group.
- When the last of its hosts in a particular multicast group leaves the group, sends an unsolicited leave group membership report to the all-routers group (244.0.0.2).

Configuring IGMP Proxy

To configure a downstream interface, enable IGMP on that interface. To configure IGMP proxy on the router, complete the following tasks:

1. Enable IP multicast.

   host1(config)#**ip multicast-routing**

2. Identify the interface that you want to act as the upstream interface.

3. Enable IGMP proxy on that interface.

   host1(config-if)#**ip igmp-proxy**

4. (Optional) Specify how often the router sends unsolicited reports to routers on the upstream interface.

   host1(config-if)#**ip igmp-proxy unsolicited-report-interval 600**

5. (Optional) Specify how long the router calculates an IGMPv1 querier router to exists on the subnetwork after the router receives an IGMPv1 query on this interface.

   host1(config-if)#**ip igmp-proxy V1-router-present-time 600**

ip igmp-proxy

- Use to enable IGMP proxy on an interface.
- The interface for which you enable IGMP proxy is the upstream interface.

NOTE: You can enable only one upstream interface.

- You can specify either IGMP proxy version 2 or 3. The default is version 2.
- Example

  host1(config)#**ip multicast-routing**
  host1(config-if)#**ip igmp-proxy**

- Use the **no** version to disable IGMP proxy on an interface.

ip igmp-proxy unsolicited-report-interval

- Use to specify the interval, in tenths of a second, at which the upstream interface transmits unsolicited reports.

NOTE: Issue this command only on the upstream interface. Otherwise, this command has no effect.

- Example

  host1(config-if)#**ip igmp-proxy unsolicited-report-interval 600**

- Use the **no** version to transmit unsolicited reports using the default value, 400 tenths of a second.

ip igmp-proxy V1-router-present-time

- Use to specify how long, in seconds, the router calculates an IGMPv1 querier router to exist on the subnetwork after the router receives an IGMP V1 query on this interface.

NOTE: Issue this command only on the upstream interface. Otherwise, this command has no effect.

- Example

  host1(config-if)#**ip igmp-proxy V1-router-present-time 600**

- Use the **no** version to set the time to the default value, 10 seconds.

Establishing the IGMP Proxy Baseline

You can set the counters for the number of queries received and reports sent on the upstream interface to zero. This feature enables you to establish a reference point, or baseline, for IGMP proxy statistics.

baseline ip igmp-proxy interface

- Use to set the counters for the number of queries received and reports sent on the upstream interface to zero.

NOTE: Issue this command only on the upstream interface. Otherwise, this command has no effect.

- Example

  (host1)#**baseline ip igmp-proxy interface**

- There is no **no** version.

Monitoring IGMP Proxy

To display IGMP proxy parameters, use the following **show** commands.

show ip igmp-proxy

- Use to display IGMP proxy parameters for a VR.
- Field descriptions
  - Routing Process—IGMP proxy protocol
  - Administrative state—State of IGMP proxy in the software: enabled or disabled

- Operational state—Operational state of IGMP proxy: enabled or disabled
- total interface—Number of IGMP proxy interfaces on the VR; currently only one upstream interface per VR
- state—Operational state of the IGMP proxy interfaces: enabled or disabled
- multicast group—Number of multicast groups associated with IGMP proxy interfaces

- Example

```
host1#show ip igmp-proxy
Routing Process IGMP Proxy, Administrative state enabled, Operational state
enabled
total 1 upstream interface, state enabled
    6 multicast group
```

show ip igmp-proxy groups

- Use to display information about multicast groups that IGMP proxy reported.
- Field descriptions
  - Grp Address—Address of the multicast group
  - Interface—Type and specifier of the upstream interface associated with the multicast group
  - Member State—State of the associated group address and interface
    - Idle—Interface is going to send a group membership report to respond to a group membership query for this group
    - Delay—Interface has responded to the latest group membership query for this group
  - count—Total number of multicast groups associated with this interface
- Example 1

```
host1#show ip igmp-proxy groups

Grp Address     Interface       Member State
--------------- --------------- --------------
225.1.1.1       atm3/0.2          Idle
225.1.1.2       atm3/0.2          Idle
225.1.1.3       atm3/0.2          Idle
225.1.1.4       atm3/0.2          Idle
225.1.1.5       atm3/0.2          Idle
225.1.1.6       atm3/0.2          Idle
count 6
```

- Example 2

```
host1#show ip igmp-proxy group 225.1.1.1
Grp Address     Interface       Member State
--------------- --------------- --------------
225.1.1.1       atm3/0.2        Idle
```

- Example 3

```
host1#show ip igmp-proxy group count
Count: 6 groups
```

show ip igmp-proxy interface

- Use to display information about the interface on which you configured IGMP proxy.
- To view information about a particular interface, enter an interface type and specifier, such as atm 3/0. For details about interface types and specifiers, see *JUNOSe Command Reference Guide, About This Guide*.
- Specify the **brief** keyword to display a summary rather than a detailed description.
- Field descriptions
  - Interface—Type of upstream interface. For details about interface types, see *JUNOSe Command Reference Guide, About This Guide*.
  - address—Address of upstream interface
  - Administrative state—State of upstream interface in the software: enabled or disabled
  - Operational state—Physical state of upstream interface: enabled or disabled
  - Version—IGMP version on this interface
  - State—Presence of IGMPv1 routers on the same subnet as this upstream interface
  - Unsolicited report interval—Time interval, in tenths of a second, at which this upstream interface sends an unsolicited group membership report
  - Version 1 router present timeout—How long, in seconds, that the upstream interface calculates an IGMPv1 router to exist on the subnet after that interface receives an IGMPv1 group membership query
  - multicast group—Number of multicast groups associated with this upstream interface
  - Interface statistics Rcvd—Statistics for messages received on this interface
    - v1 queries—Number of IGMPv1 group membership queries received
    - v2 queries—Number of IGMPv2 group membership queries received
    - v1 reports—Number of IGMPv1 group membership reports received
    - v2 reports—Number of IGMPv2 group membership reports received
  - Interface statistics Sent—Statistics for messages sent from this interface
    - v1 reports—Number of IGMPv1 leave group reports sent
    - v2 reports—Number of IGMPv2 leave group reports sent
    - leaves—Number of leave group membership messages sent
- Example

```
host1#show ip igmp-proxy interface atm 3/0.2
Interface atm3/0.2 address 21.1.1.1/255.0.0.0
Administrative state enabled, Operational state enabled
Interface parameters:
    Version 2
    State No v1 Router Present
    Unsolicited report interval 10 secs
    Version 1 router present timeout 400 secs
```

```
    0 multicast group
Interface statistics:
    Rcvd:  0 v1 query, 6 v2 queries
           0 v1 report, 0 v2 report
    Sent:  0 v1 report, 48 v2 reports, 0 leave
```

Chapter 3

Configuring PIM for IPv4 Multicast

The Protocol Independent Multicast (PIM) protocol is a collection of multicast routing protocols that enables multicast routers to identify other multicast routers to receive packets.

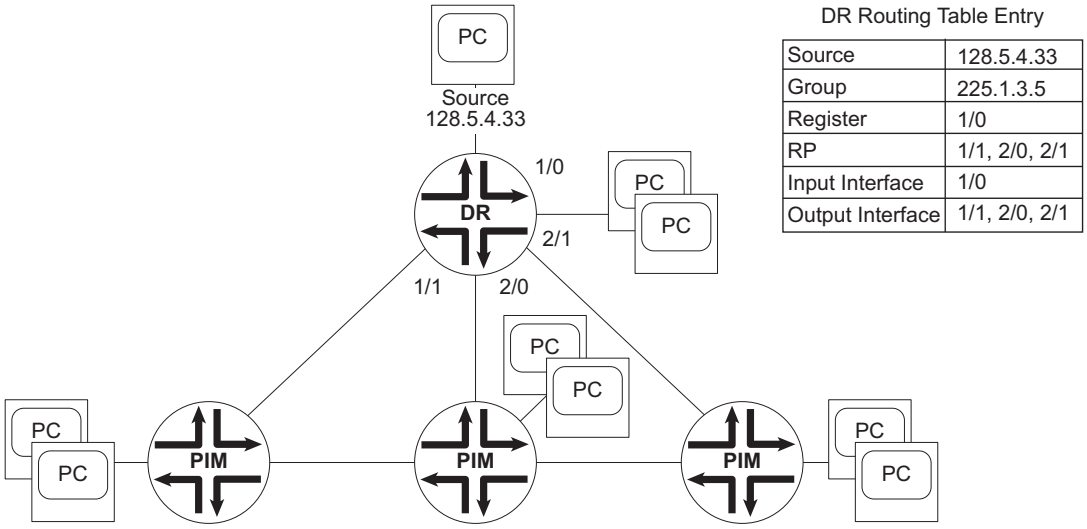
This chapter describes how to configure PIM for IPv4 on E-series routers; it contains the following sections:

Overview

The IPv4 implementation of PIM supports PIM dense mode, PIM sparse mode, PIM sparse-dense mode, and PIM source-specific multicast (PIM SSM).

Figure 7 represents how PIM builds a source, group (S,G) entry in a source-rooted tree (SRT). When multiple routers are connected to a multiaccess network, one router becomes the designated router. The designated router receives data from the source on interface 1/0 and multicasts the data to its downstream neighbors on interfaces 1/1, 2/0, and 2/1. In the designated router routing table, the entry for this operation lists the source as the IP address of the source and the group as the IP address of the multicast group.

Figure 7: Source-Rooted Tree



Neighbors exchange hello messages periodically to determine the designated router. The router with the highest network layer address becomes the designated router. If the designated router subsequently receives a hello message from a neighbor with a higher network layer address, that neighbor becomes the designated router.

PIM Dense Mode

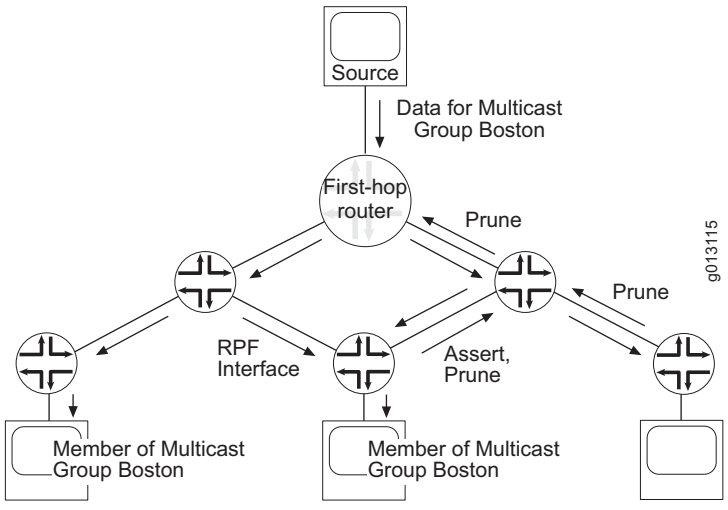
PIM dense mode uses a reverse-path multicast, flood-and-prune mechanism. The protocol was developed for situations that meet one or more of the following criteria:

- Sources and receivers are close together, and there are many more receivers than sources.
- There is a constant stream of multicast data.
- There is a lot of multicast data.

Dense-mode routing protocols use *SRT algorithms*. An SRT algorithm establishes a tree that connects each source in a multicast group to the members of the group. All traffic for the multicast group passes along this tree.

Figure 8 illustrates how PIM dense mode works. When a source sends a multicast packet to a first-hop router, the first-hop router multicasts that packet to its neighbors. Those neighbors in turn forward the packet to their neighbors and their hosts that belong to the multicast group. If a neighbor has no hosts that belong to the multicast group and has no other PIM neighbors, it returns a prune message to the first-hop router. The first-hop router does not multicast subsequent packets for that group to neighbors who respond with prune messages.

Figure 8: PIM Dense Mode Operation



Overriding Prunes

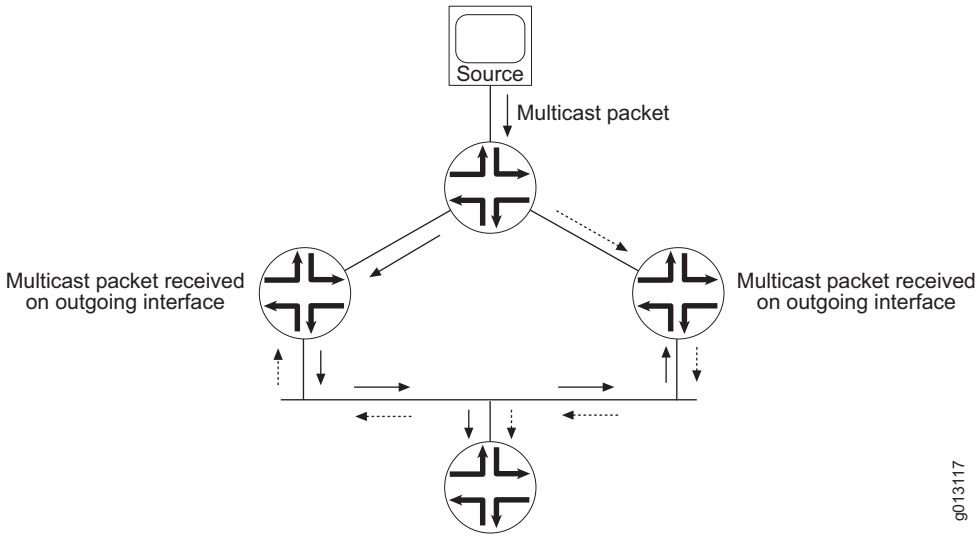
If a host on a previously pruned branch requests to join a multicast group, it sends an IGMP message to its first-hop router. The first-hop router then sends a graft message upstream.

PIM routers send join messages on multiaccess interfaces to override prune messages. For example, if a PIM router sent a prune message to indicate that it had no hosts for a multicast group, and one of its hosts subsequently requests to send a packet to that group, the router sends a join message to the first-hop router.

Preventing Duplication

If there are parallel paths to a source, duplicate packets can travel downstream through different routers to the network. If a forwarding router receives a multicast packet on its outgoing interface, the router identifies that the packet is a duplicate and notifies the upstream routers. See Figure 9.

Figure 9: Detecting Duplication



The upstream routers responsible for the duplication send assert messages to determine which router becomes the forwarder. Downstream routers listen to the assert messages to discover which router becomes the forwarder.

PIM Sparse Mode

This implementation of PIM sparse mode supports the following features:

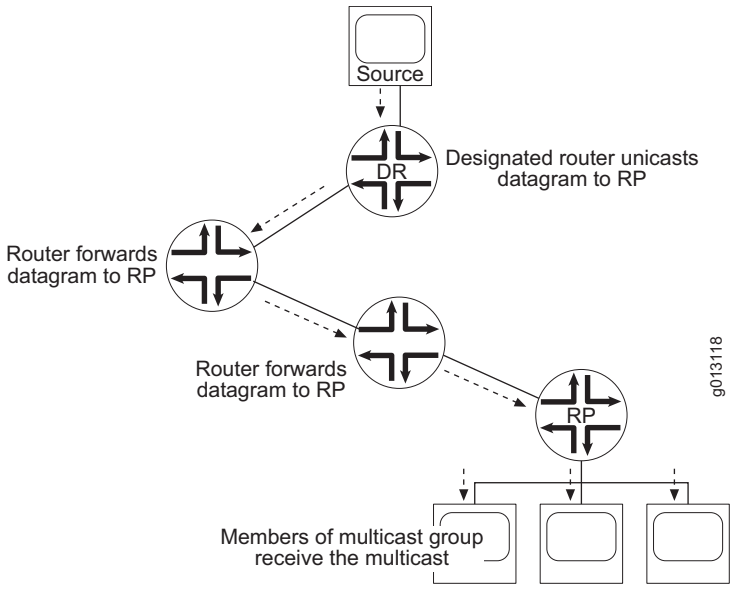
- Rendezvous point (RP) routers
- Designated routers and designated router election
- Join/prune messages, hello messages, assert messages, and register messages
- Switching from a shared tree to a shortest path tree (SPT)
- (*,*,RP) support for interoperation with dense-mode protocols
- RPF checks of multicast entries when unicast routing configuration changes
- Timers for tree maintenance
- Border, null, Rendezvous Point Tree (RPT), SPT, and wildcard flags

PIM sparse mode resolves situations that meet one or more of the following criteria:

- The multicast group contains few receivers.
- Multicast traffic is infrequent.
- Wide area networks (WANs) separate sources and receivers.

Sparse-mode routing protocols use *shared trees*. In a shared tree, sources forward multicast datagrams to a directly connected router, the designated router. The designated router encapsulates the datagram and unicasts it to an assigned RP router, which then forwards the datagram to members of multicast groups. See Figure 10.

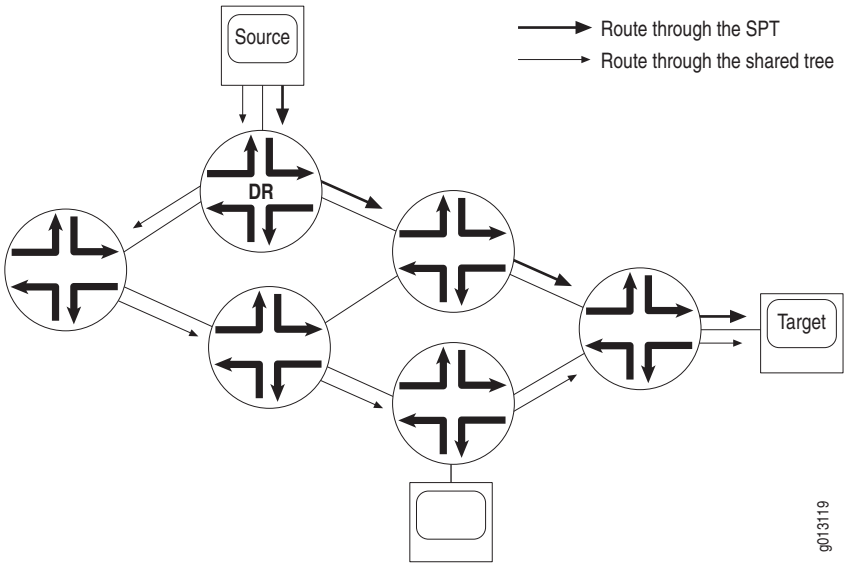
Figure 10: PIM Sparse Mode Operation



In PIM sparse mode, an RP announces a source and establishes paths from the source to members of a multicast group before multicasting any datagrams. RPs transmit join messages to become part of the shared tree that enables distribution of packets to the multicast group.

However, when a source starts multicasting datagrams, PIM sparse mode can switch to an SRT—known in PIM sparse mode as an SPT—to improve the network's efficiency. Although shared trees minimize the traffic in the network and the costs associated with unnecessary transmission of data, the routes in a shared tree might be longer than those in an SPT. See Figure 11.

Figure 11: Shared Tree Versus SPT



The designated routers on the network determine when the source switches from a shared tree to an SPT. A designated router switches to the SPT when it receives a certain number of packets which you can configure.

When all designated routers associated with a specific RP router have switched to the SPT, the RP router sends a join/prune message toward the multicast source. When the multicast source receives this message, it stops sending multicast data through the SPT.

Joining Groups

A host's designated router (DR) sends join messages to the RP when that host wants to join a group. When a host wants to leave a group, it communicates with its designated router through IGMP. When the designated router no longer has any hosts that belong to a particular group, it sends a prune message to the RP.

Timers

PIM sparse mode uses timers to maintain the networking trees.

NOTE: PIM sparse mode routers poll their neighbors and hosts for various pieces of information at set intervals.

If a PIM sparse mode router does not receive information from a neighbor or host within a specific time, known as the *hold time*, it removes the associated information from its routing tables.

You can configure how often an interface sends hello messages (hello interval) and how often routers send RP announce messages (RP announce interval). The hold-time associated with hello messages is 3.5 times the hello interval, and the holdtime associated with RP announce messages is 2.5 times the RP announce interval.

All other timers are fixed and take the default values recommended in RFC 2934—Protocol Independent Multicast MIB for IPv4 (October 2000).

PIM Sparse Mode Bootstrap Router

PIM sparse mode routers need the address of the rendezvous point (RP) for each group for which they have (*,G) state. They obtain this address either through a bootstrap mechanism or through static configuration. PIM sparse mode routers commonly use one of two bootstrap mechanisms: bootstrap router (BSR) or auto-RP. Auto-RP is standards based, but is not used in IPv6 implementations, so BSR configuration has become more popular.

When implemented, BSR operates as follows:

1. One router in each PIM domain is elected the BSR.
2. All the routers in the domain that are configured to be RP candidates periodically unicast their candidacy to the BSR.
3. The BSR picks an RP set from the available candidates and periodically announces this set in a bootstrap message.
4. Bootstrap messages are flooded hop by hop throughout the domain until all routers in the domain learn the RP set.

NOTE: A PIM router can receive group-to-RP mappings from either BSR or auto-RP, but not from both. Because BSR and auto-RP use different mapping algorithms, the mechanisms cannot coexist.

PIM Sparse-Dense Mode

In PIM sparse-dense mode, if an RP is not known for a group, the router sends data using PIM dense mode. However, if the router discovers an RP or you configure an RP statically, PIM sparse mode takes over.

You can configure both PIM dense mode and PIM sparse mode commands in PIM sparse-dense mode.

PIM Source-Specific Multicast

PIM SSM is an extension of the PIM protocol. Using SSM, a client can receive multicast traffic directly from the source. PIM SSM uses PIM sparse mode functionality to create an SPT between the client and the source, but builds the SPT without using an RP.

By default, the SSM group multicast address is limited to the IP address range 232.0.0.0 to 232.255.255.255. You can use the **ip pim ssm** command to extend SSM operations into another Class D range. (See *Configuring PIM SSM* on page 102.)

An SSM-configured network has the following advantages over a traditionally configured PIM sparse mode network include the following:

- No need for shared trees or RP mapping (no RP is required).
- No need for RP-to-RP source discovery through Multicast Source Discovery Protocol (MSDP).
- Simplified administrative deployment; you need only configure PIM sparse mode on all router interfaces and issue the necessary SSM commands (including specifying IGMPv3 on the receiver local area network).
- Support for source lists; you can use source lists, supported in IGMPv3, where only specified sources send traffic to the SSM group.

In a PIM SSM–configured network, an E-series router subscribes to an SSM channel (by means of IGMPv3 or by means of IGMP ssm-mapping for IGMPv2/v1 joins), requesting to join group G and source S. The directly connected PIM sparse mode router, the designated router of the receiver, sends an (S,G) join message to its RPF neighbor for the source. For PIM SSM, the RP is not contacted in this process by the receiver (as happens in normal PIM sparse mode operations).

Platform Considerations

For information about modules that support PIM on the ERX-7xx models, ERX-14xx models, and the ERX-310 router:

- See *ERX Module Guide, Table 1, Module Combinations* for detailed module specifications.
- See *ERX Module Guide, Appendix A, Module Protocol Support* for information about the modules that support PIM.

For information about modules that support PIM on the E120 router and the E320 router:

- See *E120 and E320 Module Guide, Table 1, Modules and IOAs* for detailed module specifications.
- See *E120 and E320 Module Guide, Appendix A, IOA Protocol Support* for information about the modules that support PIM.

References

For more information about PIM, see the following resources:

- Protocol Independent Multicast MIB for IPv4—draft-ietf-idmr-pim-mib-10.txt (July 2000 expiration)
- RFC 2362—Protocol Independent Multicast-Sparse Mode (PIM-SM): Protocol Specification (June 1998)
- RFC 3569—An Overview of Source-Specific Multicast (SSM) (July 2003)
- Source-Specific Multicast for IP—draft-ietf-ssm-arch-06.txt (March 2005 expiration)
- Source-Specific Protocol Independent Multicast in 232/8—draft-ietf-mboned-ssm232-08.txt (September 2004 expiration)
- Multicast in MPLS/BGP VPNs—draft-rosen-vpn-mcast-06.txt (April 2004 expiration)
- Multicast in MPLS/BGP IP VPNs—draft-rosen-vpn-mcast-08.txt (June 2005 expiration)

NOTE: IETF drafts are valid for only 6 months from the date of issuance. They must be considered as works in progress. Please refer to the IETF Web site at http://www.ietf.org for the latest drafts.

Before You Begin

You can configure PIM on IPv4 and IPv6 interfaces. However, IPv6 does not support all PIM configuration options. For information about configuring PIM on IPv6 interfaces, see *Chapter 7, Configuring PIM for IPv6 Multicast*.

Enabling PIM on a VR

By default, PIM is disabled. To enable PIM on a VR:

1. Enable multicast routing. (See *Enabling IP Multicast* on page 7.)
2. Create a VR, or access an existing VR context.

   host1(config)#**virtual-router boston**

3. Create and enable PIM processing.

   host1:boston(config**)#router pim**

router pim

- Use to create and enable PIM processing on a VR or to access Router Configuration mode.
- Example

  host1:boston(config)#**router pim**

- Use the **no** version to remove PIM from the VR.

Disabling PIM on a VR

To disable PIM processing on a router, use the **pim disable** command.

pim disable

- Use to disable PIM processing. By default, PIM processing is enabled.
- Example

  host1:boston(config-router)#**pim disable**

- Use the **no** version to reenable PIM processing.

Enabling PIM on an Interface

You can enable PIM on an interface in one of the PIM modes (dense, sparse, or sparse-dense) and specify how often the interface sends hello messages to neighbors.

You can configure PIM and IGMP on the same interface. If you configure IGMP and PIM on an interface, the router determines that PIM owns the interface.

NOTE: You cannot configure DVMRP and PIM on the same interface.

ip pim

- Use to enable PIM on an interface; dense mode is the default.
- Example

  host1(config-if)#**ip pim sparse-dense-mode**

- Use the **no** version to disable PIM on an interface.

ip pim query-interval

- Use to specify the interval, in seconds, at which the router sends hello messages to neighbors.
- Example

  host1(config-if)#**ip pim query-interval 100**

- Use the **no** version to restore the default setting, 30 seconds.

ip pim sparse-mode graceful-restart-duration

- Use to set the graceful restart duration for IP PIM sparse mode.
- Example

  host1(config-if)#**ip pim sparse-mode graceful-restart-duration 10**

- Use the **no** version to return to the default duration of 30 seconds.

Setting a Priority to Determine the Designated Router

You can influence whether a particular router is selected as the designated router with the **ip pim dr-priority** command. A higher priority value increases the likelihood that a router is selected as the designated router, while a lower value decreases the likelihood.

ip pim dr-priority

- Use to set a priority value, in the range 1–254, by which a router is likely to be selected as the designated router.
- Example

  host1(config-if)#**ip pim dr-priority 24**

- Use the **no** version to restore the default value, 1.

Configuring an RP Router for PIM Sparse Mode and PIM Sparse-Dense Mode

When you use the router for PIM sparse mode or PIM sparse-dense mode, some VRs must act as RP routers. You can configure static RP routers or configure the router to assign RP routers automatically.

To configure the router to assign RP routers automatically, you must define several VRs as RP routers and one VR as an RP mapping agent. RP routers send their announcement messages to the RP mapping agent, which assigns groups to RP routers and resolves any conflicts. The RP mapping agent notifies neighbors of the RP assigned to each group.

Configuring a Static RP Router

If you want to control PIM more tightly, you can configure a static RP router. To do so:

1. Configure an access list that specifies the multicast groups that can use the static RP router.

   host1(config)#**access-list boston permit 228.0.0.0 15.255.255.255**

2. Specify a static RP router.

   host1(config)#**ip pim rp-address 122.0.0.1 1 boston**

Configuring an Auto-RP Router for PIM Sparse Mode

Two multicast groups, 224.0.1.39 and 224.0.1.40, are reserved for forwarding auto-RP messages through the network. When you configure an auto-RP router for PIM sparse mode, you must assign a static RP router to these two groups. You can then specify an RP mapping agent for other multicast groups.

To configure an auto-RP router for PIM sparse mode:

1. Configure a static RP to have priority over the auto-RP for the groups that send auto-RP multicast messages.

   host1(config)#**access-list 11 permit 224.0.1.39 0.0.0.0**
   host1(config)#**access-list 11 permit 224.0.1.40 0.0.0.0**
   host1(config)#**ip pim rp-address 192.48.1.22 11 override**

2. Assign an RP mapping agent.

   host1(config)#**ip pim send-rp-discovery scope 23 loopback 1**

3. Configure routers to send auto-RP announcement messages to the mapping agent.

   host1(config)#**ip pim send-rp-announce loopback 2 scope 16 group-list 1**

Configuring an Auto-RP Router for PIM Sparse-Dense Mode

In PIM sparse-dense mode, you must prevent routers from advertising auto-RP messages to the multicast groups 224.0.1.39 and 224.0.1.40, which are reserved for forwarding auto-RP messages through the network. To configure an auto-RP router for PIM sparse-dense mode:

1. Assign an RP mapping agent.

   host1(config)#**ip pim send-rp-discovery scope 23 loopback 1**

2. Configure an access list that details the multicast groups that can use the static RP router.

   host1(config)**access-list boston permit 224.0.0.0 15.255.255.255**

3. Prevent routers from advertising auto-RP messages to the multicast groups that are reserved for forwarding auto-RP messages through the network.

   host1(config)#**access-list 1 deny 224.0.1.39**
   host1(config)#**access-list 1 deny 224.0.1.40**

4. Configure routers to send auto-RP announcement messages to the mapping agent.

   host1(config)#**ip pim send-rp-announce loopback 2 scope 23 group-list boston interval 200**

ip pim rp-address

- Use to specify a static PIM RP router.
- Specify a standard IP access list of multicast groups to control which multicast groups can use this RP router.
- Specify the **override** keyword if you want this static RP router to have priority over auto-RP routers.
- Example

  host1(config)#**ip pim rp-address 192.48.1.22 11 override**

- Use the **no** version to clear the filter from this interface.

ip pim send-rp-announce

- Use to send auto-RP announcement messages from a router you configured as an RP.
- Specify an interface type and specifier, such as atm 3/0. For details about interface types and specifiers, see *JUNOSe Command Reference Guide, About This Guide*.
- The auto-RP announcement messages contain the IP address for the interface that you specify.
- Specify the number of hops for which the announcement is valid; default value is 64.
- Specify an access list that details which multicast groups the RP can include in announcement messages.
- Specify a time interval in the range 1–65535 seconds to control how often the router sends announcements. The default is 60 seconds.
- Example

  host1(config)#**ip pim send-rp-announce loopback 2 scope 23 group-list boston interval 200**

- Use the **no** version to clear filters from this interface.

ip pim send-rp-discovery scope

- Use to configure the router as an RP mapping agent, which records group-to-RP mappings and notifies PIM designated routers about the mappings.
- Specify the number of hops for which the RP discovery message is valid; default value is 64.
- To assign an interface from which the router sends auto-RP discovery messages, specify an interface type and specifier, such as atm 3/0. For details about interface types and specifiers, see *JUNOSe Command Reference Guide, About This Guide*.
- Example

  host1(config)#**ip pim send-rp-discovery scope 23 loopback 1**

- Use the **no** version to stop the router from acting as an RP mapping agent.

Configuring BSR and RP Candidates for PIM Sparse Mode

When choosing candidate BSRs, select well-connected routers in the core of the network. Typically, candidate BSRs are a subset of the candidate RPs. A single BSR is elected for the domain of candidate BSRs. The elected BSR floods bootstrap messages (BSMs) containing their group-to-RP mappings to all PIM routers. PIM routers use the group-to-RP mappings supplied by the elected (or preferred) BSR.

ip pim bsr-candidate

- Use to define a router as a BSR candidate.
- To assign an interface from which the router sends messages, specify an interface type and specifier, such as atm 3/0. For details about interface types and specifiers, see *JUNOSe Command Reference Guide, About This Guide*.
- Specify a length (up to 32 bits) for the hash mask length field sent in BSMs that the router originates. This mask is combined with the group address before the router calls the hash function. For example, specifying a value of 24 limits the group address to the first 24 bits. The default hash mask length is 30 bits.
- Use the **priority** keyword to specify a value for the BSR-priority field of BSMs that the router originates. In the BSR election process, the BSR with the higher priority is preferred. If the priority values are equal, the router with the higher IP address becomes the BSR. The default value is 0 (address comparison only).
- Use the **period** keyword to specify the interval, in the range 1–65535 seconds, at which the BSR sends bootstrap messages. The default value is 60 seconds.
- Example

  host1(config)#**ip pim bsr-candidate loopback 1 30 10**

- Use the **no** version to stop the router from acting as a BSR candidate.

ip pim rp-candidate

- Use to define a router as an RP candidate.
- To assign an interface from which the router sends messages, specify an interface type and specifier, such as atm 3/0. For details about interface types and specifiers, see *JUNOSe Command Reference Guide, About This Guide*.
- Use the **group-list** keyword to specify an access-list that contains the set of group prefixes supported by this candidate RP (C-RP). If you do not specify a group-list, the default is the entire multicast address range.

NOTE: Because BSR has no mechanism for distributing negative entries, you should not configure negative access-list entries (also called deny access-list entries).

- Use the **hold-time** keyword to specify the amount of time the BSR keeps an RP in its C-RP list if the BSR does not receive a C-RP advertisement message. The default value is 150 seconds.
- Use the **priority** keyword to specify a priority field value that the C-RP sends to the BSR in C-RP advertisement messages. In the RP election process, the RP with the lower priority value is preferred. The default value is 192.
- Use the **interval** keyword to specify an interval, in the range 1–65535 seconds, at which the C-RP sends advertisement messages to the BSR. The default value is 60 seconds.
- Example

  host1(config)#**access-list 1 permit 227.0.0.0 15.255.255.255**
  host1(config)#**access-list 1 permit 228.0.0.0 15.255.255.255**
  host1(config)#**ip pim rp-candidate loopback 1 group-list 1**

- Use the **no** version to stop the router from acting as an RP candidate

Migrating to BSR from Auto-RP

Migrating to BSR from auto-RP requires that you upgrade all PIM routers in the domain to support BSR. However, until all routers are BSR-capable, continue to use auto-RP.

After all routers are BSR-capable, switch from auto-RP to BSR as follows:

1. Use the **no ip pim send-rp-discovery scope** command to stop PIM in the network by disabling all auto-RP mapping agents. This results in flooding to an empty map.

2. Reconfigure auto-RP mapping agents as candidate BSRs by using the **ip pim bsr-candidate** command.

3. Reconfigure auto-RP candidate RPs as BSR candidate RPs by issuing the **no ip pim send-rp-announce** command and then issuing the **ip pim rp-candidate** command.

Switching to an SPT for PIM Sparse Mode

PIM sparse mode initiates multicasting using a shared tree. You can configure PIM sparse mode to switch to an SPT when a source starts sending multicast messages, or you can prevent PIM sparse mode from switching to an SPT. Multicasting over an SPT might be more efficient than multicasting over a shared tree. (See *PIM Sparse Mode*, earlier in this chapter.)

ip pim spt-threshold

- Use to specify when PIM sparse mode switches from a shared tree to an SPT.
- Specify a nonzero integer or the keyword **infinity** to prevent PIM sparse mode from switching to an SPT.
- Specify a value of 0 (default) to configure PIM to switch to an SPT when a source starts sending multicast messages.
- Example

  host1(config)#**ip pim spt-threshold 4**

- Use the **no** version to restore the default value, 0.

Creating Multicast VPNs

JUNOSe router software provides the ability to create multicast VPNs by using GRE tunnels. This implementation is based on *Multicast in MPLS/BGP VPNs* (draft-rosen-vpn-mcast-06.txt and draft-rosen-vpn-mcast-08.txt) and further defined by *Base Specification for Multicast in MPLS/BGP VPNs* (draft-raggarwa-13vpn-2547-mvpn-00.txt).

NOTE: Although you can configure PIM sparse mode remote neighbors, you can no longer use these remote neighbors for BGP/MPLS VPNs. For multicast VPNs, use the functionality described in this section.

The JUNOSe software supports default Multicast Distribution Trees (MDTs) and data MDTs.

Creating Multicast VPNs Using the Default MDT

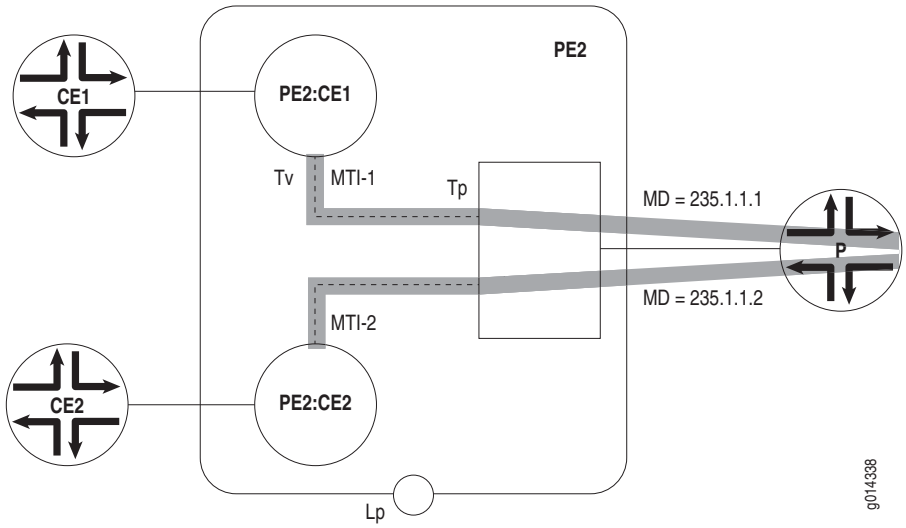
The JUNOSe software does not support a single MDT command. Instead, you must configure the multicast tunnel interfaces (MTIs) explicitly. The MTI is an IP interface that is stacked on a GRE tunnel interface. The destination address of the GRE tunnel is the multicast VPN (MVPN) group address of the MDT.

A **tunnel mdt** command specifies that the tunnel is the MTI for the default MDT, enabling the creation of a second, layer 2 interface (interface tunnel gre:*name*.mdt) on which an unnumbered IP interface (tied to the provider edge loopback interface) is stacked in the context of the parent virtual router.

Multicast VPN Configuration Example

In the following example (Figure 12), customer edge router 1 (CE1) and customer edge router 2 (CE2) exist in two separate VPNs. Each VPN is configured with its assigned Multicast Domain (235.1.1.1 and 235.1.1.2, respectively).

Figure 12: Multicast VPNs



To better understand the example, keep the following in mind:

- Lp is a loopback interface in the parent router. This address is the loopback interface used as the BGP peer address of the provider edge router (PE). Its address is advertised in the provider address space.

- Tv is the MTI in the VRF. This interface is typically configured as a PIM sparse-mode interface (though you can configure it for dense-mode or sparse-dense-mode). Any packets that originate in the VRF are sent using the address of this interface as the source address. You must set this interface address to be identical to loopback interface of the parent router (Lp).

CAUTION: Defining the Tv interface with an address other than the loopback interface of the parent router might restrict operation with non-Juniper Networks routers.

- Tp is an unnumbered IP interface that is tied to the loopback interface of the provider edge router (PE).

To configure the example, use the following general procedures:

NOTE: This example provides general information for configuring a simple Multicast VPN network. For detailed information about creating GRE tunnels, see *JUNOSe IP Services Configuration Guide, Chapter 10, Configuring IP Tunnels*. For detailed information about PIM sparse-mode configuration, see *PIM Sparse Mode* on page 80.

1. Configure BGP/MPLS VPN.

   ```
   host1:PE2(config-router)#router bgp 100
   host1:PE2(config-router)#address-family vpnv4 unicast
   host1:PE2(config-router-af)#neighbor 1.1.1.1 activate
   host1:PE2(config-router-af)#neighbor 1.1.1.1 next-hop-self
   host1:PE2(config-router-af)#neighbor 3.3.3.3 activate
   host1:PE2(config-router-af)#neighbor 3.3.3.3 next-hop-self
   host1:PE2(config-router-af)#exit-address-family
   ```

   See *JUNOSe BGP and MPLS Configuration Guide, Chapter 3, Configuring BGP-MPLS Applications* for details.

2. Configure PIM sparse mode in the core and RP for MVPN group addresses.

NOTE: For MVPN, it is a typical practice to use shared trees.

```
host1:PE1(config-router)#virtual-router PE2
host1:PE2(config)#ip multicast-routing
host1:PE2(config)#
host1:PE2(config)#! MDT RP is 72.72.72.72 (P1)
host1:PE2(config)#access-list 1 permit ip 235.0.0.0 0.255.255.255 any
host1:PE2(config)#ip pim rp-address 72.72.72.72 1
host1:PE2(config)#
host1:PE2(config)#! Do not switch from RPT for MDTs
host1:PE2(config)#ip pim spt-threshold infinity group-list 1
host1:PE2(config)#
```

3. Configure the loopback interface, Lp, in parent router PE2.

   ```
   host1:PE2(config)#interface loopback 0
   host1:PE2(config-if)#ip address 2.2.2.2 255.255.255.255
   host1:PE2(config-if)#ip pim sparse-mode
   host1:PE2(config-if)
   ```

NOTE: You must configure the loopback interface for PIM sparse mode to support unnumbered MDTs.

4. Add PIM-SM to core-facing interfaces.

   ```
   host1:PE2(config)#interface atm2/1.20
   host1:PE2(config-subif)#ip pim sparse-mode
   host1:PE2(config-subif)#
   ```

5. Extend the BGP router configuration to contribute VPN routes into the multicast router table of the VRF using the **ip route-type both** command.

```
host1:PE2(config)#router bgp 100
host1:PE2(config-router)#address-family ipv4 unicast vrf PE21
host1:PE2(config-router-af)#ip route-type both
host1:PE2(config-router-af)#exit
host1:PE2(config-router)#
```

6. Configure the GRE tunnel for VPN1.

```
host1(config)#interface tunnel gre:MTI-21 transport-virtual-router PE2
host1(config-if)#tunnel source 2.2.2.2
host1(config-if)#tunnel destination 235.1.1.1
host1(config-if)#tunnel mdt
host1(config-if)#exit
host1(config)#
```

7. Configure the GRE tunnel for VPN2

```
host1(config)#interface tunnel gre:MTI-22 transport-virtual-router PE2
thost1(config-if)#tunnel source 2.2.2.2
host1(config-if)#tunnel destination 235.1.1.2
host1(config-if)#tunnel mdt
host1(config-if)#exit
host1(config)#
```

8. Configure the IP interface (Tv) in PE2:CE1 as a PIM sparse-mode interface with the address of the loopback interface.

```
host1(config)#virtual-router PE2:CE21
host1:PE2:CE21(config)#interface tunnel gre:MTI-21
host1:PE2:CE21(config)#ip address 2.2.2.2 255.255.255.255
host1:PE2:CE21(config)#ip pim sparse-mode
host1:PE2:CE21(config)#exit
host1:PE2:CE21#
```

9. Configure the IP interface (Tv) in PE2:CE2 as a PIM sparse-mode interface with the address of the loopback interface (same as the loopback 0 address for PE2).

```
host1:PE2:CE21(config)#interface loopback 0
host1:PE2:CE21(config-if)#ip address 2.2.2.2 255.255.255.255
host1:PE2:CE21(config-if)#exit
host1:PE2:CE21(config)#exit
host1:PE2:CE21#virtual-router PE2:CE22
host1:PE2:CE22#configuration terminal
host1:PE2:CE22(config)#interface tunnel gre:MTI-22
host1:PE2:CE22(config)#ip unnumbered loopback 0
host1:PE2:CE22(config)#ip pim sparse-mode
host1:PE2:CE22(config)#exit
host1:PE2:CE22#
```

10. Configure the Tp interfaces as unnumbered IP interfaces.

```
host1(config)#interface tunnel gre:MTI-21.mdt
host1(config-if)#ip unnumbered loopback 0
host1(config-if)#ip pim sparse-mode
```

```
host1(config-if)#exit
host1(config)#

host1(config)#interface tunnel gre:MTI-22.mdt
host1(configif)#ip unnumbered loopback 0
host1(config-if)#ip pim sparse-mode
host1(config-if)#exit
host1(config)#
```

tunnel mdt

- Use to enable multicast distribution tree operation so the IP tunnel component can create an MDT interface. This command functions for GRE interfaces only.
- Example

  ```
  host1(config-if)#tunnel mdt
  ```

- The **no** version disables MDT on the interface.

Creating Multicast VPNs Using the Data MDT

A data multicast distribution tree (MDT), based on section 8 of Internet draft draft-rosen-vpn-mcast-08.txt, *Multicast in MPLS/BGP IP VPNs*, solves the problem of P routers flooding unnecessary multicast information to PE routers that have no interested receivers for a particular VPN multicast group. The data MDT solution requires the creation of a new tunnel by the PE router if the source exceeds a configured rate threshold parameter. All other PE routers join the new tunnel only if the PE router has receivers in the VPN for that multicast group.

The JUNOSe software uses dynamic point-to-multipoint GRE tunnels to configure data MDTs. In the current release, IPv6 transport over GRE (unicast or multicast) is not supported. For more information, see *JUNOSe IP Services Configuration Guide, Chapter 11, Configuring Dynamic IP Tunnels*.

Data MDTs are established using PIM-SM (shared RP Trees) and PIM-SSM (Source Trees). Profiles for dynamic interfaces in the VRF are restricted to sparse-mode only.

Data MDT Sources

A C-SG flow arriving in the source VRF is a candidate for a data MDT if the system matches the C-SG in the route map that you specify for the data MDT using the **ip pim data-mdt** command. The C-SG flow is initially forwarded on the default MDT. The system creates the data MDT when the flow rate exceeds a value you configure in the route map using the **set threshold** command.

When the Source C-PIM-SM first creates a data MDT for a C-SG flow, it sends a <C-SG, P-G> MDT join message with type, length, value (TLV) format to the default MDT. This message invites peer PE routers to join the new data MDT. It starts a timer that you can configure using the **mdt-data-delay** command to track the number of seconds before switching to the data MDT. When that timer expires, C-PIM-SM switches from sending C-SG data on the default MDT to sending data on the data MDT.

When the C-SG flow is switched to the data MDT, the Source C-PIM-SM starts a timer that you can configure using the **mdt-data-holddown** command to track the number of seconds before switching to the default MDT. When the timer expires, the data MDT is deleted and the C-SG flow switched back to the default MDT if the flow rate drops back below the threshold. If the flow rate exceeds the threshold, the timer restarts. If the timer expires and the flow rate is below the threshold, the data MDT is removed.

The Source C-PIM-SM maintains sent MDT Join TLV messages in its database as long as they are active. While the data MDT is active, C-PIM-SM resends that MLD Join TLV message using a setting that you can configure using the **mdt-interval** command to measure time in seconds between successive MLD join TLV messages.

Data MDT Receivers

When the Receiver C-PIM-SM receives a < C-SG, P-G > MDT Join TLV message from the default MDT, it extracts the C-SG and the data MDT P-Group address from the TLV and queries the route map that you specified for the data MDT to determine whether the C-SG is a candidate for a data MDT. If it matches, the C-PIM-SM adds the MDT Join TLV to its database and records the time.

If the Receiver C-PIM-SM does not receive an MDT Join TLV < C-SG, P-G > to refresh its database within the amount of time specified for the timeout in the **mdt-data-timeou**t command, the MDT Join TLV < C-SG > is removed from the database and the associated data MDT is removed.

When a new MDT Join TLV < C-SG, P-G > is added to the database, the Receiver C-PIM-SM determines whether it has an SG, SPT state. If it has an SG state, and the incoming interface (IIF) is the default MDT, then C-PIM-SM creates the data MDT and deletes the corresponding forwarding entry. C-PIM-SM waits for the source to transmit data on the data MDT. During this period, data can continue to be received on the default MDT. C-PIM-SM fails the reverse-path forwarding (RPF) check, which results in a forwarding entry with a discarded IIF.

If the C-SG,SPT state is created (either as a result of a C-SSM join or switch from RPT to SPT), and it is the default MDT, the Receiver C-PIM-SM determines whether an MDT Join TLV < C-SG > is active. If it is, C-PIM-SM creates the data MDT.

Establishing a Data MDT Using ASM or SSM

A data MDT carries one C-SG flow. If the data MDTs are established using any-source multicast (ASM), then the P-Group address selected by a PE for the data MDT must be unique to that PE in the MDT (that is, the range of MDT P-Group addresses available in the core must be administratively divided among all the PEs that will source VPN multicasts). The VRFs in a PE must share the P-Group addresses in the assigned range for the PE.

If the data MDTs are established using single-source multicast (SSM), you must configure VRFs to transmit on a tunnel using the same MDT P-Group address. Each VRF transmits using a unique P-Source address; however, each data MDT created by the VRF must use a different P-Group address. There might be one sender data MDT and possibly many receiver data MDTs sharing an IP tunnel. Each PE can assign MDT P-Groups from the same range, but the P-Group addresses must be administratively divided among the VPNs.

For a receiver on the data MDT, P-PIM-SM joins the data MDT by propagating join state into the core. The P-Group for that join is extracted from the MDT Join TLV. If SSM is not activated or the P-Group is not in the SSM group range, P-PIM-SM performs a <*, G> join towards the RP for that P-Group.

If SSM is activated and the P-Group is in the SSM group range, P-PIM-SM performs an <S, G> join towards the P-Source, where the P-Source address is the SA of the MDT Join TLV.

Configuring Data MDTs

To configure data MDTs:

1. Configure a dynamic interface profile to specify the PIM configuration of the IP/MTI interface in the VRF.

   ```
   host1(config)#profile pe13DataMdtMti
   host1(config-profile)#ip virtual-router "pe1:pe13"
   host1(config-profile)#ip unnumbered loopback 0
   host1(config-profile)#ip pim sparse-mode
   ```

2. Configure a dynamic interface profile to specify the IP/MDT interface in the parent.

   ```
   host1(config-profile)#profile pe1DataMdtMdt
   host1(config-profile)#ip virtual-router pe1
   host1(config-profile)#ip unnumbered loopback 0
   host1(config-profile)#ip pim sparse-mode
   ```

3. Configure the destination profile for dynamic IP tunnel creation.

   ```
   host1(config-profile)#gre destination profile pe13DataMdtProfile
   virtual-router pe1
   host1(config-dest-profile)#tunnel destination subnet 233.3.0.0 255.255.0.0
   host1(config-dest-profile)#tunnel source 1.1.1.1
   host1(config-dest-profile)#tunnel mdt profile pe1DataMdtMdt
   host1(config-dest-profile)#profile pe13DataMdtMti
   host1(config-dest-profile)#virtual-router pe1
   ```

   For more information about creating dynamic IP tunnels, see *JUNOSe IP Services Configuration Guide, Chapter 11, Configuring Dynamic IP Tunnels*.

4. Configure the VRF, including an access list to match <S,G> and <*,G> entries.

   ```
   host1:pe1(config)#ip vrf pe13
   host1:pe1(config-vrf)#rd 100:13
   host1:pe1(config-vrf)#route-target both 100:3
   host1:pe1(config-vrf)#interface tunnel gre:MTI-13.mdt
   host1:pe1(config-if)# ip unnumbered loopback 0
   host1:pe1(config-if)# ip pim sparse-mode
   host1:pe1(config-if)#access-list pe13DataMdt permit ip any 225.1.0.0
   0.0.255.255
   ```

5. Specify a route map to configure the set of (S, G) for which data MDTs can be created, and the threshold to be applied for each SG.

   host1:pe1(config)#**route-map pe13MdtThresholds permit 10**
   host1:pe1(config-route-map)#**match ip address pe13DataMdtSend**
   host1:pe1(config-route-map)#**set threshold 0**
   host1:pe1(config-route-map)#**route-map pe13MdtThresholds permit 20**
   host1:pe1(config-route-map)#**match ip address pe13DataMdt**

6. Configure the group address pools in the route map.

   host1:pe1(config-route-map)#**ip pim group-address-pool pe13DataMdtGroups 233.3.1.0 233.3.1.255**

   If the data MDTs are established using ASM, you must divide the range of available MDT P-Group addresses so that PEs source VPN multicasts. All VRFs in a PE draw from a single address pool that contains the range of group addresses assigned to that PE.

   If the data MDTs are established using SSM, you can configure VRFs to transmit on a tunnel using the same MDT P-Group address. Each VRF transmits using a unique P-Source address; however, each data MDT created by the VRF must use a different P-Group address. There might be one sender data MDT and possibly many receiver data MDTs sharing an IP tunnel.

   For SSM, each PE can assign MDT P-Groups from the same range, but the P-Group addresses must be administratively divided among the VPNs.

7. Configure the tunnel for the VRF.

   host1:pe1(config)#**virtual-router pe1:pe13**
   host1:pe1:pe13(config)#**interface tunnel gre:MTI-13 transport-virtual-router pe1**
   host1:pe1:pe13(config)#**tunnel source 1.1.1.1**
   host1:pe1:pe13(config)#**tunnel destination 235.3.3.3**
   host1:pe1:pe13(config)#**tunnel mdt**
   host1:pe1:pe13(config)#**ip unnumbered loopback 0**
   host1:pe1:pe13(config)#**ip pim sparse-mode**

8. Configure the data MDT.

   host1:pe1:pe13(config)#**ip pim data-mdt**
   host1:pe1:pe13(config-ip-pim-data-mdt)#**tunnel source 1.1.1.1**
   host1:pe1:pe13(config-ip-pim-data-mdt)#**tunnel group-address-pool pe13DataMdtG$**
   host1:pe1:pe13(config-ip-pim-data-mdt)#**route-map pe13MdtThresholds**

ip pim

- Use to enable PIM on an interface.
- Example

  host1(config-if)#**ip pim sparse-dense-mode**

- Use the **no** version to disable PIM on an interface.

ip pim data-mdt

- Use to activate data MDTs and enter IP PIM Data MDT Configuration mode.
- Example

  host1(config)#**ip pim data-mdt**

- Use the **no** version to deactivate data MDTs.

ip pim group-address-pool

- Use to configure PIM group address pools from which data MDT group addresses are allocated.
- Example

  host1(config)#**ip pim group-address-pool pe21DataMDT 232.1.0.0 232.2.255.255**

- There is no **no** version.

mdt-data-delay

- Use to configure a delay before switching to data MDT.
- The delay is measured by 0.1 seconds; the default is 30.
- Example

  host1(ip-pim-data-mdt-config)#**mdt-data-delay 20**

- Use the **no** version to return to the default.

mdt-data-holddown

- Use to configure the time in seconds before switching to the default MDT group from the data MDT group.
- The default is 60.
- Example

  host1(ip-pim-data-mdt-config)#**mdt-data-holddown 200**

- Use the **no** version to return to the default.

mdt-data-timeout

- Use to configure the time in seconds before the flow leaves the data MDT group.
- The default is 180.
- Example

  host1(ip-pim-data-mdt-config)#**mdt-data-timeout 160**

- Use the **no** version to return to the default.

mdt-interval

- Use to configure the time in seconds between successive MLD join TLV messages.
- The default is 60.
- Example

  host1(ip-pim-data-mdt-config)#**mdt-interval 80**

- Use the **no** version to return to the default.

set threshold

- Use to configure a threshold value for multicast VPN applications, including default MDT and data MDT.
- Example

  host1(config)#**set threshold 30**

- Use the **no** version to remove the threshold.

tunnel group-address-pool

- Use to configure a group address pool for a data MDT tunnel.
- Example

  host1(ip-pim-data-mdt-config)#**tunnel group-address-pool dataMDT1**

- Use the **no** version to delete the group address pool.

Using PIM Sparse Mode Join Filters

You can use PIM sparse mode join filters to prevent multicast state from being created in the PIM sparse mode router. The filters are applied to join entries in PIM join/prune messages that are received from PIM sparse mode neighbors.

By denying joins at the edge of a network, you can limit the multicast state and traffic in the network. By accepting only certain joins, you can control which multicast services an end user can receive. PIM join filters also reduce the potential for denial of service (DoS) attacks where large numbers of joins forwarded to each router on the RPT can result in a PIM state explosion and very high memory consumption.

For information about how to create access lists, see *JUNOSe IP Services Configuration Guide, Chapter 1, Configuring Routing Policy*.

ip pim join-filter

- Use to specify an extended access list that you want this PIM interface to use as a join filter.
- You can apply the join filter at the global level or at the interface level.
- If an interface-level filter exists, it takes precedence over the global-level filter.

- Example 1

  host1(config)#**ip pim join-filter gold**

- Example 2

  host1(config-interface)#**ip pim join-filter gold**

- Use the **no** version to remove the filter association.

Configuring PIM SSM

Source Specific Multicast (SSM) is a datagram delivery model that best supports one-to-many applications, also known as broadcast applications. SSM is networking technology that targets audio and video broadcast application environments.

To configure PIM SSM, you enable PIM SSM on the router and define the SSM range of IP multicast addresses.

To use PIM SSM, IGMPv3 must be configured on customer premise equipment (CPE)–facing interfaces to receivers, and PIM sparse mode must be configured on CPE-facing interfaces to sources and on core-facing interfaces. After configuring SSM, you can use the **show ip pim sparse-mode sg-state** command to display SSM group membership information.

To configure PIM SSM:

1. Enable PIM SSM on the E-series router. The IANA SSM range is configured by default. You can modify the SSM address range by using the access list.

   host1(config)#**access-list 15 permit ip any host 239.0.0.2**
   host1(config)#**access-list 15 permit ip any 232.0.0.0 0.225.225.225**
   host1(config)#**ip pim ssm range 15**

2. Enable PIM sparse mode on the CPE-facing interface towards the source or core.

3. Enable IGMPv3 on the CPE-facing interface towards the receiver. PIM SSM also works with IGMPv2 if you configure the ssm-map in IGMP as in the following example:

PIM SSM also works with IGMPv2 if you configure the ssm-map in IGMP as in the following example:

host1(config)#**ip pim ssm**
host1(config)#**access-list ssm_map1 permit 232.0.0.1 255.255.255.255**
host1(config)#**ip igmp ssm-map enable**
host1(config)#**ip igmp ssm-map static ssm_map1 51.0.0.1**

The **no** version disables ssm-map:

host1(config)#**no ip igmp ssm-map static ssm_map1 51.0.0.1**

ip pim ssm

- Use to enable PIM SSM and define the SSM range for IPv4 multicast addresses.
- Use the **range** keyword to define the SSM range of IP multicast addresses.
- Example 1—Enables SSM with addresses in the IANA range. The SSM address range is set as the default and by default, the SSM group multicast address is limited to the IPv4 address range 232.0.0.0/6.

  host1(config)#**ip pim ssm**

- Example 2—Configures Class D addresses outside of the default SSM range.

  host1(config)#**access-list alist permit any 223.0.0.0 0.255.255.255**
  host1(config)#**ip pim ssm range alist**

- Example 3—Resets the SSM address range to the default.

  host1(config)#**ip pim ssm default**

- Use the **no** version to disable SSM.

Configuring the BFD Protocol for PIM

The **ip pim bfd-liveness-detection** command configures the Bidirectional Forwarding Detection (BFD) protocol for PIM. The BFD protocol uses control packets and shorter detection time limits to more rapidly detect failures in a network. Also, because they are adjustable, you can modify the BFD timers for more or less aggressive failure detection.

PIM routers send periodic hello messages from each PIM-enabled interface. You can configure this interval using the **ip pim query-interval** command. By default, the PIM router sends a hello message every 30 seconds (with an interval range of 0–210 seconds). If it receives no response from a neighbor within 3.5 times the interval value (a minimum of 3.5 seconds), the PIM router drops the neighbor.

In contrast, when a BFD session exists between neighbors, a PIM neighbor that goes down is detected quickly (in milliseconds rather than in seconds).

When you issue the **ip pim bfd-liveness-detection** command on a PIM router, the router establishes BFD liveness detection with all BFD-enabled PIM neighbors. When the local router receives an update from a remote PIM neighbor—if BFD is enabled and if the session is not already present—the local router attempts to create a BFD session to the remote neighbor.

Each adjacent pair of neighbors negotiates an acceptable transmit interval for BFD packets. The negotiated value can be different on each neighbor. Each neighbor then calculates a BFD liveness detection interval. When a neighbor does not receive a BFD packet within the detection interval, it declares the BFD session to be down.

NOTE: Before the router can use the **ip pim bfd-liveness-detection** command, you must specify a BFD license key. To view an already configured license, use the **show license bfd** command.

For general information about configuring and monitoring the BFD protocol, see *JUNOSe IP Services Configuration Guide, Chapter 5, Configuring BFD*.

ip pim bfd-liveness-detection

- Use to enable BFD (bidirectional forwarding detection) and define BFD values to more quickly detect PIM data path failures.
- The neighbors in a PIM network use the configured values to negotiate the actual transmit intervals for BFD packets.
  - You can use the **minimum-transmit-interval** keyword to specify the interval at which the local router proposes to transmit BFD control packets to its neighbors. The default value is 300 milliseconds.
  - You can use the **minimum-receive-interval** keyword to specify the minimum interval at which the local router must receive BFD control packets from its neighbors. The default value is 300 milliseconds.
  - You can use the **minimum-interval** keyword to specify the same value for both of those intervals. Configuring a minimum interval has the same effect as configuring the minimum receive interval and the minimum transmit interval to the same value. The default value is 300 milliseconds.
- You can use the **multiplier** keyword to specify the detection multiplier value. The calculated BFD liveness detection interval can be different on each neighbor. The multiplier value is roughly equivalent to the number of packets that can be missed before the BFD session is declared to be down. The default value is 3.
- For details on liveness detection negotiation, see *Negotiation of the BFD Liveness Detection Interval* section in *JUNOSe IP Services Configuration Guide, Chapter 5, Configuring BFD*.
- You can change the BFD liveness detection parameters at any time without stopping or restarting the existing session; BFD automatically adjusts to the new parameter value. However, no changes to BFD parameters take place until the values resynchronize with each neighbor.
- Example

  host1(configif)#**ip pim bfd-liveness-detection minimum-interval 800**

- Use the **no** version to disable BFD on the PIM interface.

Removing PIM

To remove PIM from a VR, use the **no router pim** command.

router pim

- Use to create and enable PIM processing on a VR or to access Router Configuration mode.
- Example

  host1:boston(config)#**router pim**

- Use the **no** version to remove PIM from the VR.

Resetting PIM Counters and Mappings

You can use the **clear ip pim** commands to reset PIM counters and mappings.

clear ip pim auto-rp

- Use to clear the group-to-RP router mappings that the router learned through auto-RP.
- Specify the IP address of an RP to clear the group-to-RP mappings for a particular RP. If you do not specify an IP address, the router clears the group-to-RP mappings on all RP routers learned through auto-RP.
- Example

  host1#**clear ip pim auto-rp 192.34.56.7**

- There is no **no** version.

clear ip pim interface count

- Use to clear the counters for multicast packet statistics on all interfaces or a specified interface.
- Specify an interface type and specifier, such as atm 3/0, to clear the counters on that interface. For details about interface types and specifiers, see *JUNOSe Command Reference Guide, About This Guide*.
- If you do not specify an interface, the router clears the counters on all interfaces.
- Example

  host1#**clear ip pim interface atm 3/0.5 count**

- There is no **no** version.

Monitoring PIM

You can display information about PIM events and parameters.

Monitoring PIM Events

You can use the debug PIM commands to view information about PIM events.

debug ip pim

- Use to show information about the selected event.
- To control the type of events displayed, specify a severity level.
- To control how much information to display, specify a verbosity level.
- Example

  host1#**debug ip pim events severity 1 verbosity low**

- Use the **no** version to disable the display.

undebug ip pim

- Use to turn off the display of information previously enabled with the **debug ip pim** command.
- Example

  host1#**undebug ip pim events**

- There is no **no** version.

Monitoring PIM Settings

You can use the **show ip pim** commands to display information about PIM settings.

show ip pim

- Use to view general PIM router-level information.
- Field descriptions
  - Default PIM Version—Default PIM version number (always 2)
  - Default Domain Id—Default Domain Id (always 0)
  - Default Hello period—Default interval (in minutes) at which the router sends hello messages to neighbors
  - Default Hello Hold Time—Default time (in minutes) for which the router keeps the neighbor state alive
  - Default J / P Hold Time—Hold time value (in seconds) set in Join/Prune messages originated by this PIM router
  - Keepalive Period—Time SG join state is maintained in the absence of SG Join message
  - Assert Time—Period after last assert before assert state is timed out
  - Register Suppression Time—Period during which a designated router stops sending registers to the RP

- Register Probe Time—Time before register suppression time (RST) expires when a designated router might send a NULL-Register to the RP
- Register TTL—TTL value (in PIM register packets) originated by this PIM router
- SSM—State of SSM on this PIM router (enabled or disabled)
- range—Default SSM group range or name of the access list specifying the range
- Sparse-Mode Graceful Restart Duration—Restart interval in seconds
- Join filter—Name of the join filter access-list (if configured) for this PIM router

- Example

```
host1:1#show ip pim
Default PIM Version: 2
Default Domain Id: 0
Default Hello Period: 30
Default Hello HoldTime: 105
Default J/P HoldTime: 210
Keepalive Period: 210
Assert Time: 210
Register Suppression Time: 60
Register Probe Time: 5
Register TTL: 64
SSM enabled, range default
Sparse-Mode Graceful Restart Duration: 30
Graceful restart is complete (timer 0 seconds)
Join filter, access-list bronze
```

show ip pim auto-rp

- Use to display information about RP routers and the RP mapping agent in a PIM sparse mode environment.
- Field descriptions
  - Configured with ttl—Number of hops for which the RP discovery message is valid
  - Using interface addr—IP address of the interface from which the router sends RP discovery messages
  - interval—Time interval, in seconds, at which the router sends RP discovery messages
  - PIM AutoRP candidate RP mapping(s)—Routers that the RP mapping agent is evaluating to determine an RP router for this interface
- Example 1

```
host1:1#show ip pim auto-rp
This PIM router is an Auto RP mapping agent.
  Configured with ttl 64
  [ Using interface addr 121.0.0.1, interval 60 ].
PIM AutoRP candidate RP mapping(s)
```

- Example 2

```
host1:1#show ip pim auto-rp
This PIM router is _not_ an Auto RP mapping agent.
PIM AutoRP candidate RP mapping(s)
Candidate RP 122.0.0.1
  Group(s) 224.0.0.0/4, AutoRP, ttl 64, interval 60, from access List 1
Candidate RP 122.0.0.1
  Group(s) 224.0.1.39/32 (negative), AutoRP, ttl 64, interval 60, from access List 1
Candidate RP 122.0.0.1
  Group(s) 224.0.1.40/32 (negative), AutoRP, ttl 64, interval 60, from access List 1
```

show ip pim bsr

- Use to display BSR information and the group prefixes for which the local router is a candidate RP in a PIM sparse mode environment.
- Field descriptions
  - Candidacy—Whether the router is a candidate BSR
  - Configured on—Interface on which the router is configured
  - address—Address of the router
  - hashMaskLen—Hash mask length
  - priority—Priority of the router
  - period—Time between bootstrap messages, in seconds
  - Elected BSR—This router or IP address of the elected bootstrap router
  - next BSM—If BSR is this router, time until the next bootstrap message is sent, in seconds
  - expires in—If BSR is not this router, time until the elected BSR expires if no bootstrap messages are received
  - Local candidate RP mapping(s)—Routers that the mapping agent is evaluating to determine an RP router for this interface
- Example 1—On a router that is the elected BSR

```
host1:1#show ip pim bsr

This PIM router is a Candidate BSR.
  Configured on intf ATM3/0.101, address: 101.0.0.1
  hashMaskLen 30, priority 2, period 60 seconds.
Elected BSR is this router, next BSM in 3 seconds.

Local candidate RP mapping(s):
Candidate RP 101.0.0.1
  224.0.0.0/4, BSR, hold-time 150, interval 60, priority 192
  228.0.0.0/24, BSR, hold-time 150, interval 60, priority 192, from
access-list acl
  230.0.0.0/24, BSR, hold-time 150, interval 60, priority 192, from
access-list acl
```

- Example 2—On a router that is a candidate BSR

```
host1:1#show ip pim bsr
This PIM router is a Candidate BSR.
  Configured on intf ATM3/0.100, address: 100.0.0.1
  hashMaskLen 30, priority 2, period 60 seconds.
Elected BSR is 101.0.0.1 (priority 0), expires in 73 seconds.
```

- Example 3—On a router that is not a candidate BSR

```
host1:1#show ip pim bsr
This PIM router is not a Candidate BSR.
Elected BSR is 101.0.0.1 (priority 0), expires in 73 seconds.
```

show ip pim data-mdt

- Use to display information about active data MDTs.
- To display information about data MDTs on which the provider edge transmits data, use the **sender** keyword.
- To display information about data MDTs on which the provider edge receives data, use the **receiver** keyword.
- To display information about an IP PIM group address pool, use the **group** keyword.
- To display a summary of configuration for each data MDT, use the **summary** keyword.
- To display the number of data MDTs, use the **count** keyword.
- Field descriptions
  - PE *Name*—Name of the PE
  - C-SG—Address of the C-SG
  - P-SG—Address of the P-SG
  - MTI—Name of the dynamic IP tunnel on which the data MDT was created
  - Data rate/Threshold—Rate and threshold of multicast data
  - Time until next MDT Join TLV—Configured delay until next MDT Join TLV
  - Time until MDT Join TLV expires—Configured delay until MDT Join TLV expires
  - Time until switchover from default-MDT—Configured delay until the data MDT switches over to the default MDT
- Example 1—Displays information about a data MDT sender

```
host1:PE1#show ip pim data-mdt 225.1.1.1
PE11 - Sender
  C-SG: 10.11.0.100, 225.1.1.1
  P-SG: 1.1.1.1, 235.0.1.1
  MTI: TUNNEL gre:mvpn-dynamic-1
    Data rate/Threshold: 10012/500 Kbps
    Time until next MDT Join TLV: 25 seconds
```

- Example 2—Displays information about a data MDT receiver

```
host1:PE1#show ip pim data-mdt 225.2.2.2
PE31 - Receiver
  C-SG: 10.13.0.100, 225.2.2.2
  P-SG: 3.3.3.3, 235.0.1.1
  MTI: TUNNEL gre:mvpn-dynamic-3
    Time until MDT Join TLV expires: 29 seconds
```

- Example 3—Displays a summary of data MDT senders

```
host1:PE1#show ip pim data-mdt senders summary

VRF   S/R      C-Group    C-Source     P-Group   P-Source         MTI
----   -------   ---------   -----------   ---------  --------
---------------
PE11   Sender   225.1.1.1   10.11.0.100    235.0.1.1  1.1.1.1    TUNNEL
gre:mvpn-dynamic-1
PE12   Sender   225.1.1.1   10.12.0.100    235.0.1.2  1.1.1.1    TUNNEL
gre:mvpn-dynamic-2

Counts: 2 senders, 0 receivers, total 3.
```

- Example 4—Displays the number of data MDT senders and receivers

```
host1:PE1#show ip pim data-mdt count
Counts: 2 senders, 1 receivers, total 3.
```

show ip pim dense-mode sg-state

- Use to display information for each (Source, Group) pair for PIM dense mode.
- Field descriptions
  - (Source, Group) pair—IP addresses of multicast source and group
  - EntryExpires—Time until the (S,G) pair entry expires
  - RPF Route—Reverse-path forwarding route
  - IIF—IP address of incoming interface
  - UpNbr—IP address of upstream neighbor
  - Pruned Oifs—Outgoing interfaces that have been pruned
    - Address—IP address of outgoing interface
    - IfId—Index of the interface
    - Pruned due to—Reason for prune: assert or explicit prune
    - Pruned time remaining—Time in seconds until the prune expires
- Example

```
host1:8#show ip pim dense-mode sg-state
PIM DM route table and pruned oif information
<122.0.0.1, 224.0.1.39>   EntryExpires: 99
  RPF Route: 122.0.0.0/255.0.0.0   IIF: 107.0.8.4   UpNbr: 107.0.4.8
  Pruned Oifs:
   Address: 108.0.8.5   IfId: 95
    Pruned due to assert
    Pruned time remaining 129
<130.0.0.2, 224.0.1.39>   EntryExpires: 100
  RPF Route: 130.0.0.0/255.0.0.0   IIF: 107.0.8.4   UpNbr: 107.0.4.8
```

```
  Pruned Oifs:
   Address: 108.0.8.5   IfId: 95
    Pruned due to assert
    Pruned time remaining 130
<121.0.0.1, 224.0.1.40>   EntryExpires: 102
  RPF Route: 121.0.0.0/255.0.0.0   IIF: 107.0.8.4   UpNbr: 107.0.4.8
  Pruned Oifs:
   Address: 108.0.8.5   IfId: 95
    Pruned due to assert
    Pruned time remaining 133
```

show ip pim interface

- Use to display information about PIM interfaces.
- Specify no keywords or variables to view information about all PIM interfaces.
- Provide an interface type and specifier (such as atm 3/0) to display information about that interface only. For details about interface types and specifiers, see *JUNOSe Command Reference Guide, About This Guide*.
- Specify the **summary** keyword to view the number of configured, enabled, and disabled PIM dense mode, PIM sparse mode, and PIM sparse-dense mode interfaces.
- Specify the **count** keyword to view the number of multicast packets that the interface has sent and received.
- Field descriptions
  - Interface Addr—IP address of the interface
  - Interface Name—Type and specifier of the interface. For details about interface types and specifiers, see *JUNOSe Command Reference Guide, About This Guide*.
  - Ver—Version of PIM running on this interface
  - Mode—PIM mode running on this interface: Sparse, Dense, or SparseDense
  - Nbr Count—Number of neighbors connected to this interface
  - Hello Intvl—Time interval, in seconds, at which the interface sends hello messages to neighbors
  - DR Addr—Address of the designated router
  - SM—Number of PIM sparse mode interfaces
  - DM—Number of PIM dense mode interfaces
  - SM/DM—Number of PIM sparse-dense mode interfaces
    - enabled—Number of interfaces administratively enabled
    - disabled—Number of interfaces administratively disabled
  - ControlPktCount In|Out—PIM messages received on and sent from this interface
    - Hello—Total number of hello messages
    - JoinPrune—Total number of join and prune messages
    - Assert—Total number of assert messages

- Example 1

```
host1#show ip pim interface
Interface                                          Nbr    Hello                          BFD
  Addr         Interface      State   Ver    Mode    count   Intvl   DR Addr   JoinFilter Enabled
---------   ---------------   -----   ---   ------   -----   -----   -------   ----------   -----
1.1.1.2     FastEthernet1/1   up      2     Sparse   1       30      1.1.1.2   ---          yes


Interface                                          Nbr    Hello                          BFD
  Addr         Interface      State   Ver    Mode    count   Intvl   DR Addr  JoinFilter  Enabled
---------   ---------------   -----   ---   ------   -----   -----   -------  ----------   ------
1.1.1.2     FastEthernet1/1   up      2     Sparse   1       30      1.1.1.2   ---          no
```

- Example 2

```
host1#show ip pim interface summary
PIM Interface Summary
  SM:    0, 0 enabled, 0 disabled
  DM:    0, 0 enabled, 0 disabled
  SM/DM: 1, 0 enabled, 1 disabled
```

- Example 3

```
host1#show ip pim interface count
PIM Interface Count
Interface Addr  Interface Name       ControlPktCount In|Out
                                     Hello     JoinPrune  Assert
192.32.10.20    ATM3/0.20             0         0          0
                                      0         0          0
```

show ip pim neighbor

- Use to display information about PIM neighbors that the router discovered.
- Provide an interface type and specifier (such as atm 3/0) to display information about that interface only. For details about interface types and specifiers, see *JUNOSe Command Reference Guide, About This Guide*.
- Field descriptions
  - Neighbor Addr—IP address of the neighbor
  - Interface Name—Type and specifier of the interface to which the neighbor connects. For details about interface types and specifiers, see *JUNOSe Command Reference Guide, About This Guide*.
  - Uptime—Time since the router discovered this neighbor in *days hours:minutes:seconds* format
  - Expires—Time available for the neighbor to send a hello message to the interface. If the neighbor does not send a hello message during this time, it no longer is a neighbor.
  - Ver—Version of PIM that the neighbor is running
  - Mode—PIM mode that the neighbor is using: Sparse, Dense, or SparseDense
  - BFD—BFD status: up or down

- Example

```
host1#show ip pim neighbor
Neighbor
  Addr     Interface Name     Uptime    Expires    Ver    Mode    BFD
--------   ---------------   --------   --------   ---   ------   ---
1.1.1.1    FastEthernet1/1   00:02:49   00:01:27   2     Sparse

Neighbor
  Addr     Interface Name     Uptime    Expires    Ver    Mode    BFD
--------   ---------------   --------   --------   ---   ------   ---
1.1.1.1    FastEthernet1/1   00:03:16   00:01:30   2     Sparse   up

Neighbor
  Addr     Interface Name     Uptime    Expires    Ver    Mode    BFD
--------   ---------------   --------   --------   ---   ------   ----
1.1.1.1    FastEthernet1/1   00:00:07   00:01:39   2     Sparse   down
```

show ip pim rp

- Use to display information about PIM group-to-RP mappings.
- Specify the address of a group to view PIM group-to-RP mappings for a particular group.
- Specify the mapping keyword to display all group-to-RP mappings that the router has recorded.
- Field descriptions
  - Group(s)—Prefix of the multicast group
  - RP—IP address of RP router for the multicast group
  - priority—Priority of the router
  - via—Method by which the RP router was assigned: AutoRP, Static RP, or BSR
  - expiryTime—Time in seconds at which the RP mapping becomes invalid, unless the mapping agent (access list) reassigns the RP router to this group
- Example 1

```
host1:8#show ip pim rp mapping
PIM Group-to-RP mapping(s)
Group(s) 224.0.0.0/4
  RP 122.0.0.1,  priority 0,  via  AutoRP, expiryTime 88
Group(s) 224.0.1.39/32 (negative)
  RP 122.0.0.1,  priority 0,  via  AutoRP (Negative), expiryTime 88
Group(s) 224.0.1.40/32 (negative)
  RP 122.0.0.1,  priority 0,  via  AutoRP (Negative), expiryTime 88
```

- Example 2

```
host1:8#show ip pim rp mapping
PIM Group-to-RP mapping(s)
Group(s) 224.0.0.0/4
  RP 134.0.0.1, priority 0, via Static, from access-list 1
Group(s) 232.0.0.0/16
  RP 134.0.0.1, priority 0, via BSR, expires in 121 seconds
```

show ip pim rp-hash

- Use to show which RP router that a multicast group is using.
- Field descriptions
  - Group(s)—Multicast group or groups
  - RP—RP router for the multicast group
  - priority—Priority of the router
  - via—Method by which the RP router was assigned: AutoRP, Static RP, or BSR
  - expiryTime—Time in seconds at which the RP mapping becomes invalid, unless it is renewed by the mapping agent
- Example 1

```
host1:2#show ip pim rp-hash 232.1.1.1
Group(s) 224.0.0.0/4
  RP 122.0.0.1,  priority 0,  via  AutoRP, expiryTime 128
```

- Example 2

```
host1:2#show ip pim rp-hash 226.0.0.1
Group(s) 226.0.0.0/24
  RP 101.0.0.1, priority 192, via BSR, expires in 145 seconds
 *RP 145.0.0.3, priority 192, via BSR, expires in 145 seconds
```

show ip pim sparse-mode sg-state

- Use to display information for each (S,G) pair for PIM sparse mode and PIM SSM.
- Field descriptions
  - (S, G) pair—Source, Group pair for which information is provided
  - Group-to-RP mapping—IP addresses and network mask of the multicast group
  - RP—IP address of RP router
  - SSM group—Indicator that this is an SSM group
  - RPF Route—IP address and network mask of the RPF route
  - IIF—IP address of the incoming interface for the RPF route
  - UpNbr—IP address of the upstream neighbor
  - Oifs—Outgoing interface
  - Auto RP Discovery SELF oif—Indicates that RP router for this group was assigned through auto-RP
  - Register Oif to RP—IP address of RP router for the outgoing interface; suppressed for SSM
  - Address—IP address of outgoing interface
  - Interface—Type and specifier of the interface. For details about interface types and specifiers, see *JUNOSe Command Reference Guide, About This Guide*.

- Joined as—Type of mapping
  - (S, G)—Mapping from a specific source to a specific group
  - (*, G)—Mapping from any source to a specific group
  - (*, * , RP)—Mapping from any source to any group
- Join expires—Number of seconds before the (S,G) membership expires
- Count of entries—Total counts of (S,G) pair mappings

- Example

```
host1:2#show ip pim sparse-mode sg-state
PIM SM route table and oif information
<*, 224.0.1.40>
  Group-to-RP mapping: 224.0.0.0/240.0.0.0   RP: 123.0.0.1
  RPF Route: 123.0.0.0/255.0.0.0   IIF: 106.0.7.3   UpNbr: 106.0.3.7
  Oifs:
   Auto RP Discovery SELF oif.
    Joined as <*, G>

<*, 225.1.2.3>
  Group-to-RP mapping: 224.0.0.0/240.0.0.0   RP: 123.0.0.1
  RPF Route: 123.0.0.0/255.0.0.0   IIF: 106.0.7.3   UpNbr: 106.0.3.7
  Oifs:
   Address: 78.7.7.7   Interface: loopback7
    Local group membership present.
<*, 235.1.1.1>
  Group-to-RP mapping: 224.0.0.0/240.0.0.0   RP: 123.0.0.1
  RPF Route: 123.0.0.0/255.0.0.0   IIF: 106.0.7.3   UpNbr: 106.0.3.7
  Oifs:
   Address: 78.7.7.7   Interface: loopback7
    Local group membership present.

<118.1.33.34, 232.0.0.1>
  SSM Group
  RPF Route: 118.1.0.0/255.255.0.0   IIF: 118.1.0.1 (Directly attached)
  Oifs:
   Register Oif to RP: 141.0.0.2 suppressed for SSM Group.
   Address: 134.0.0.1   Interface: ATM3/0.104
    Joined as <S, G>   Join Expires: 161

<118.1.33.35, 232.0.0.1>
  SSM Group
  RPF Route: 118.1.0.0/255.255.0.0   IIF: 118.1.0.1 (Directly attached)
  Oifs:
   Register Oif to RP: 141.0.0.2 suppressed for SSM Group.
   Address: 134.0.0.1   Interface: ATM3/0.104
    Joined as <S, G>   Join Expires: 161

<10.0.1.8, 235.1.1.1>      EntryExpires: 143
  Group-to-RP mapping: 224.0.0.0/240.0.0.0   RP: 123.0.0.1
  RPF Route: 10.0.0.0/255.0.0.0   IIF: 106.0.7.3   UpNbr: 106.0.3.7
  Oifs:
   Address: 78.7.7.7   Interface: loopback7
    Joined as <*, G>

Count of entries - <S, G>    : 3
                   <*, G>    : 3
                   <*, *, RP>: 0
```

show ip pim sparse-mode unicast-route

- Use to display the unicast routes that PIM sparse mode is using.
- Field descriptions
  - Route—IP address and network mask for the unicast route
  - RpfNbr—RPF neighbor
  - Iif—Incoming interface for the unicast route
  - Pref—Preference value for the unicast route
  - Metric—Value of metric for the unicast route (type of metric varies with the unicast protocol)
  - Count of entries—Number of unicast routes that PIM sparse mode is using
- Example

```
host1:2#show ip pim sparse-mode unicast-route
PIM SM unicast route table information
Route                          RpfNbr          Iif             Pref  Metric
------------------------------------------------------------------------------
122.0.0.0      /255.0.0.0                       122.0.0.1       255   1
Count of entries: 1
```

show ip pim spt-threshold

- Use to display the threshold for switching to the shortest path tree at a PIM designated router.
- Field descriptions
  - Access List Name—Name of the IP access list that specifies the groups to which the threshold applies
  - SptThreshold (in kbps)—Value at which PIM sparse mode switches from a shared tree to an SPT. A value of infinity indicates that PIM sparse mode never switches to an SPT.
- Example

```
host1:2#show ip pim spt-threshold
Access List Name                SptThreshold(in kbps)
------------------------------------------------------
1                               infinity
```

Chapter 4
Configuring DVMRP

E-series routers support Distance Vector Multicast Routing Protocol (DVMRP) on VRs to forward multicast datagrams through a network. DVMRP is an interior gateway protocol that supports operations within an autonomous system, but not between autonomous systems. The multicast backbone of the Internet, MBone, uses DVMRP to forward multicast datagrams. This chapter describes how to configure DVMRP on E-series routers; it contains the following sections:

Overview

NOTE: PIM has gained general acceptance among a large number of multicast-enabled networks. We recommend that you use PIM rather than DVMRP for applications that are not otherwise required to run DVMRP.

DVMRP is a dense-mode multicasting protocol and therefore uses a broadcast and prune mechanism. The protocol builds a source-rooted tree (SRT) in a similar way to PIM dense mode. DVMRP routers flood datagrams to all interfaces except the one that provides the shortest unicast route to the source. DVMRP uses pruning to prevent unnecessary sending of multicast messages through the SRT.

A DVMRP router sends prune messages to its neighbors if it discovers that:

- The network to which a host is attached has no active members of the multicast group.
- All neighbors, except the next-hop neighbor connected to the source, have pruned the source and the group.

When a neighbor receives a prune message from a DVMRP router, it removes that neighbor from its (S,G) pair table, which provides information to the multicast forwarding table.

When a host on a previously pruned branch attempts to join a multicast group, it sends an IGMP message to its first-hop router. The first-hop router then sends a graft message upstream.

Identifying Neighbors

In this implementation of DVMRP, a *neighbor* is a directly connected DVMRP router. When you enable DVMRP on an interface, the associated VR adds information about local networks to its DVMRP routing table. The VR then sends probe messages periodically to learn about neighbors on each of its interfaces. To ensure compatibility with other DVMRP routers that do not send probe messages, the VR also updates its DVMRP routing table when it receives route report messages from such routers.

Advertising Routes

As its name suggests, DVMRP uses a distance-vector routing algorithm. Such algorithms require that each router periodically inform its neighbors of its routing table. DVMRP routers advertise routes by sending DVMRP report messages. For each network path, the receiving router picks the neighbor advertising the lowest cost and adds that entry to its routing table for future advertisement.

The cost, or metric, for this routing protocol is the hop count back to the source. The hop count for a network device is the number of routers on the route between the source and that network device.

Table 7 shows an example of the routing table for a DVMRP router.

Table 7: Sample Routing Table for a DVMRP Router

Source Subnet	Subnet Mask	From Router	Metric	Time Before Entry Is Deleted from Routing Table	Input Port	Output Port
143.2.0.0	255.255.0.0	143.32.44.12	4	85	3/0	4/0, 4/1
143.3.0.0	255.255.0.0	143.2.55.23	2	80	3/1	4/0, 4/1
143.4.0.0	255.255.0.0	143.78.6.43	3	120	3/1	4/0, 4/1

The DVMRP router maintains an (S,G) pair table that provides information to the multicast forwarding table. The (S,G) pair table is based on:

- Information from the DVMRP routing table
- Information learned from prune messages
- If IGMP and DVMRP are on the same interface, group information learned from IGMP

The (S,G) pair table includes a route from each subnetwork that contains a source to each multicast group of which that source is a member. These routes can be static or learned routes. Table 8 shows an example of the (S,G) pair table for DVMRP.

Table 8: Sample DVMRP (S,G) Pair Table

Source Subnet	Multicast Group	Time Before Entry Is Deleted from Routing Table	Input Port	Output Port
143.2.0.0	230.1.2.3	85	3/0	4/0, 4/1
	230.2.3.4	75	3/0	4/0, 4/1
	230.3.4.5	60	3/0	4/1
	230.4.5.6	90	[a]	4/0
143.3.0.0	230.1.2.3	80	3/1	4/0, 4/1

a. No value for the input port indicates that the interface is associated with a protocol other than DVMRP.

Platform Considerations

For information about modules that support DVMRP on the ERX-7xx models, ERX-14xx models, and the ERX-310 router:

- See *ERX Module Guide, Table 1, Module Combinations* for detailed module specifications.
- See *ERX Module Guide, Appendix A, Module Protocol Support* for information about the modules that support DVMRP.

For information about modules that support DVMRP on the E120 router and the E320 router:

- See *E120 and E320 Module Guide, Table 1, Modules and IOAs* for detailed module specifications.
- See *E120 and E320 Module Guide, Appendix A, IOA Protocol Support* for information about the modules that support DVMRP.

References

For more information about DVMRP, see the following resource:

- Distance Vector Multicast Routing Protocol—draft-ietf-idmr-dvmrp-v3-11.txt (April 2004 expiration)

Before You Begin

You can configure multicasting on IPv4 and IPv6 interfaces.

For information about configuring IP interfaces, see *JUNOSe IP, IPv6, and IGP Configuration Guide, Chapter 1, Configuring IP*. For information about configuring IPv6 interfaces, see *JUNOSe IP, IPv6, and IGP Configuration Guide, Chapter 2, Configuring IPv6*.

For information about configuring multicast on IPv6 interfaces, see *Chapter 5, Configuring IPv6 Multicast*.

Enabling DVMRP on a VR

By default, DVMRP is enabled on the router. To enable DVMRP on a VR:

1. Enable multicast routing.

   host1(config)#**ip multicast-routing**

2. (Optional) Create a VR or access a VR context.

   host1(config)#**virtual-router boston**

NOTE: If you do not specify a VR, you can configure DVMRP on the default router.

You must enable and configure DVMRP on one or more interfaces for DVMRP to function. See *Activating DVMRP on an Interface* on page 121. You can also set DVMRP limits for the VR; see *Configuring DVMRP Limits* on page 121.

Activating DVMRP on an Interface

By default, DVMRP is not activated on an interface. Configuring any DVMRP parameter on an interface automatically activates DVMRP on that interface. You can also activate DVMRP on an interface and use the default parameters.

ip dvmrp

- Use to activate DVMRP on an interface.
- This command automatically creates and enables DVMRP processing on the current VR.
- Issuing this command identifies this interface as one that DVMRP owns.
- Example

  host1:boston(config-if)#**ip dvmrp**

- Use the **no** version to remove DVMRP from an interface.

Configuring DVMRP Limits

You can configure DVMRP and IGMP on the same interface. If you configure DVMRP and IGMP on an interface, the router determines that DVMRP owns the interface.

NOTE: You cannot configure DVMRP and PIM on the same interface.

When you have enabled DVMRP processing on a VR, you can configure the following settings for that VR:

- The number of routes that the VR advertises on each interface.
- A maximum number of DVMRP routes at which the router generates a system log warning message and an SNMP trap.

ip dvmrp route-hog-notification

- Use to set the number of DVRMP routes that the router can record before it generates a system log warning message.
- The warning alerts you so you can identify routers that are injecting large numbers of routes into the MBone.
- Example

  host1:boston(config)#**ip dvmrp route-hog-notification 5000**

- Use the **no** version to restore the default value, 10,000 routes.

ip dvmrp route-limit

- Use to limit the number of routes that the router can advertise on each interface.
- Example

  host1:boston(config)#**ip dvmrp route-limit 5000**

- Use the **no** version to restore the default value, 7000 routes.

Filtering DVMRP Reports

You can configure an interface to accept only reports with routes that appear on a standard IP access list. You can refine the set of accepted routes further, by defining a second access list of neighbors who can supply the specified routes.

For example, suppose you define an access list that specifies that the router accepts only reports for the route 172.16.2.0/24. You then define a second access list that specifies that only neighbors 192.168.1.1 and 193.168.1.1 can supply this route. If neighbor 192.168.2.2 supplies the route, the DVMRP router rejects this report.

You can also modify the value (distance) that the router associates with a DVMRP route when it computes the RPF interface for the source of a multicast packet. By default, the router associates a distance of 0 with DVMRP routes; this value specifies that the router use DVMRP, rather than a unicast routing protocol, to transport multicast datagrams.

However, in a configuration where PIM discovers multicast routes and a unicast routing protocol performs RPF lookups, you can increase the administrative distance to favor the unicast protocol.

For information about defining access lists, see *JUNOSe IP Services Configuration Guide, Chapter 1, Configuring Routing Policy*.

ip dvmrp accept-filter

- Use to filter routes in DVMRP reports in accordance with a standard IP access list.
- Specify a standard IP access list of sources for which the interface can accept routes.
- Specify a DVMRP administrative distance to favor a unicast routing protocol.
- Specify a neighbor list to restrict the neighbors from which reports for routes on the first list can be accepted.
- Example

  host1:boston(config-if)#**ip dvmrp accept-filter boston-list 4 neighbor-list boston-neighbors**

- Use the **no** version to disable a filter.

Configuring DVMRP Summary Addresses

You can configure an interface to advertise a summary address with a known metric rather than a more specific route. DVMRP advertises the summary address if the DVMRP routing table contains a more specific route that matches the address and mask of the summary address.

If you want to advertise all routes rather than a summary, disable automatic summarization on the interface (**no ip dvmrp auto-summary**). By default, the router automatically summarizes DVMRP routes. DVMRP automatic summarization maps a unicast subnet route to a classful network number route when the subnet has a different network number from the IP address of the interface (or tunnel) over which the advertisement travels. If the interface is unnumbered, the router compares the network number of the numbered interface to the IP address to which the unnumbered interface points.

If you configure a summary address on an interface and do not disable automatic summarization, the interface advertises the least-specific address.

ip dvmrp auto-summary

- Use to reenable the router to summarize routes automatically for this interface. By default, automatic summarization is enabled.
- Example

  host1:boston(config-if)#**ip dvmrp auto-summary**

- Use the **no** version to disable automatic summarization for this interface.

ip dvmrp summary-address

- Use to advertise DVMRP summary addresses on an interface. By default, an interface advertises only summary addresses generated by automatic summarization.
- If you configure multiple overlapping summary addresses on an interface, the one with the shortest mask takes preference.
- Use the **metric** keyword to specify a DVMRP metric (hop count); the default metric value is 1.
- Example

  host1:boston(config-if)#**ip dvmrp summary-address 192.48.1.2 255.255.255.0 metric 1**

- Use the **no** version to stop advertising a summary address on the interface.

Changing the Metric for a Route

The metric for DVMRP is hop count. For example, a route with two hops over a slow serial line is preferable to a route with three hops over a faster optical line.

The router increases the number of DVMRP routes in incoming reports by a default metric of one and in outgoing reports by a default of 0. You can change the metric for an interface to promote or demote the preference for associated routes.

ip dvmrp metric-offset

- Use to adjust the number of hops associated with a route. This action specifies that the route is more efficient or less efficient than an alternative route.
- Use the **in** keyword to specify the number of hops by which the router increases the number of DVMRP routes advertised in incoming DVMRP reports. This option is the default.
- Use the **out** keyword to specify the number of hops by which the router increases the number of DVMRP routes advertised in outgoing DVMRP reports.
- Example

  host1:boston(config-if)#**ip dvmrp metric-offset in 3**

- Use the **no** version to revert to the default settings: 1 for incoming reports and 0 for outgoing reports.

Importing Routes from Other Protocols

You can import routing information from other protocols into the DVMRP routing table. Only routes that appear in the RPF table can be imported. To do so:

1. If you want to use IS-IS, OSPF, or RIP routes, make those routes available to multicast protocols. See *Defining Static Routes for Reverse-Path Forwarding* on page 7.
2. Access Router Configuration mode.
3. Specify a route map.
4. Import information from one type of routing domain into another.

redistribute

- Use to import information from another type of routing domain to the DVMRP domain. DVMRP can import only routes that appear in the RPF table.
- Specify the source protocol from which routes are being redistributed. It can be one of the following keywords: **bgp**, **isis**, **ospf**, **static**, or **connected**.
- Use the **static** keyword to redistribute static IP multicast routes into DVMRP.
- Use the **connected** keyword to redistribute routes that are established automatically in the RPF table when another multicast routing protocol, such as PIM, is enabled on an interface.

- Use the **route-map** keyword to configure the route map to filter imported routes from the source routing protocol to the current routing protocol. If you do not specify the **route-map** option, all routes are redistributed. If you specify the **route-map** option, but no route map tags are listed, no routes are imported.
- Example—Importing routing information from BGP into DVMRP

  host1:boston(config-router)#**redistribute bgp 100 route-map boston-map**

- Use the **no** version to disable redistribution.

route-map

- Use to specify a route map.
- Example

  host1:boston(config-router)#**route-map boston-map atm 3/2**

- Use the **no** version to delete the route map. If you do not specify an interface, it removes the global route map if one exists.

router dvmrp

- Use to create and enable DVMRP processing on a VR or to access DVMRP Router Configuration mode.
- Example

  host1:boston(config)#**router dvmrp**

- Use the **no** version to remove DVMRP from the VR.

Specifying Routes to Be Advertised

By default, if DVMRP owns an interface, that interface advertises all DVMRP routes it has learned to its neighbors. You can specify the routes that the interface advertises by issuing the **ip dvmrp announce-filter** command in conjunction with a standard IP access list. The IP access list defines the DVMRP routes that are advertised.

ip dvmrp announce-filter

- Use to specify the DVMRP routes for an interface to advertise.
- Specify a standard IP access list of routes for the interface to advertise.
- Example

  host1:boston(config-if)#**ip dvmrp announce-filter boston-list**

- Use the **no** version to enable the interface to advertise all DVMRP routes that it has learned.

Preventing Dynamic Route Distribution

By default, if you make changes to a route map, the router dynamically redistributes the routes in DVMRP. To prevent this dynamic redistribution, use the **disable-dynamic-redistribute** command.

disable-dynamic-redistribute

- Use to halt the dynamic redistribution of routes that are initiated by changes to a route map; dynamic redistribution is enabled by default.
- Example

  host1(config-router)#**disable-dynamic-redistribute**

- There is no **no** version.

Exchanging DVMRP Unicast Routes

DVMRP maintains its own unicast routing table, based on distance vector calculations. The routing table defines the best-known distance to each destination and how to get there. The router updates the table by exchanging information with its neighbors. The DVMRP routing table is used solely for RPF lookups.

By default, if DVMRP owns an interface, that interface exchanges DVMRP unicast routes with its neighbors, and you cannot disable the exchange of routes. However, you can enable and disable the exchange of DVMRP unicast routes on interfaces that DVMRP does not own.

When an interface exchanges DVMRP routes, the router obtains routes from DVMRP report messages and stores them in its DVMRP routing table. Other multicast protocols, such as PIM, can then use these routes for RPF lookups. With this feature, PIM can use the DVMRP routing table even when the router is not running DVMRP.

All interfaces, including tunnels, support DVMRP unicast routing. DVMRP tunnels use DVMRP multicast routing to support DVMRP unicast routing.

ip dvmrp unicast-routing

- Use to enable the exchange of DVMRP unicast routes on an interface not owned by DVMRP.
- Example

  host1:boston(config-if)#**ip dvmrp unicast-routing**

- Use the **no** version to disable the exchange of DVMRP unicast routes on an interface not owned by DVMRP.

Disabling and Removing DVMRP

You can disable DVMRP on a VR or an interface without removing the configuration. You can also remove DVMRP from a VR or an interface.

disable

- Use to disable DVMRP processing on a VR without removing the DVMRP configuration. By default, DVMRP processing is enabled.
- Example

  host1:boston(config-router)#**disable**

- Use the **no** version to reenable DVMRP processing on a VR.

ip dvmrp

- Use to activate DVMRP on an interface.
- This command automatically creates and enables DVMRP processing on the current VR.
- Issue this command to identify this interface as one that DVMRP owns.
- Example

  host1:boston(config-if)#**ip dvmrp**

- Use the **no** version to remove DVMRP from an interface.

ip dvmrp disable

- Use to disable DVMRP processing on an interface without removing the DVMRP configuration.
- Example

  host1:boston(config-if)#**ip dvmrp disable**

- Use the **no** version to reenable DVMRP processing on an interface.

router dvmrp

- Use to create and enable DVMRP processing on a VR or to access Router Configuration mode.
- Example

  host1:boston(config)#**router dvmrp**

- Use the **no** version to remove DVMRP from a VR.

Clearing DVMRP Routes

You can clear one or more routes from the DVMRP routing table. However, if you do so, the routes might reappear in the routing table if they are rediscovered.

clear ip dvmrp routes

- Use to clear DVMRP routes from the routing table.
- If you do not specify any options, the router removes all routes except those associated with its own interfaces from the DVMRP table.
- If you specify an IP address but not a subnet mask, the router removes the longest route to that IP address from the DVMRP table.
- If you specify a subnet mask, the router removes that specific route from the DVMRP table.
- Example

  host1:boston#**clear ip dvmrp routes**

- There is no **no** version.

Configuring DVMRP Tunnels

DVMRP tunnels enable the exchange of IP multicast traffic between routers separated by networks that do not support multicast routing. For information about DVMRP tunnels, see *JUNOSe IP Services Configuration Guide, Chapter 10, Configuring IP Tunnels*.

Monitoring DVMRP

You can establish a reference point for DVMRP statistics by setting the statistics counters to zero.

You can display DVMRP information with the **show ip dvmrp** commands.

baseline ip dvmrp

- Use to set the counters for DVMRP statistics to zero, which establishes a reference point, or baseline, for DVMRP statistics.
- Example

  (host1)#**baseline ip dvmrp**

- There is no **no** version.

show ip dvmrp

- Use to display DVMRP information for a VR.
- Field descriptions
  - Dvmrp Administrative State—State of DVMRP in the software: Enabled or Disabled
  - Mcast Administrative State—State of multicasting in the software: Enabled or Disabled
  - Dvmrp Version—Version of DVMRP with which this software is compatible
  - GenerationID—A number the router generates each time it reboots; when the number changes, neighbors discard all information previously learned from the router
  - Number of Routes—Number of routes in the DVMRP routing table
  - Number of Triggered Routes—Number of routes waiting to be advertised, because a parameter for the route changed
  - Reachable Routes—Number of routes that the router can currently reach
  - Route-hog-notification—Number of DVMRP routes that the router can record before it generates a system log warning message
  - Route-limit—Maximum number of routes that the router can advertise on each interface
  - Send-S32-Prunes-Only—Indicator of whether the router sends only S-32 prunes
    - true—Router sends only S-32 prunes and grafts to ensure compatibility with other protocols, such as PIM
    - false—Router sends S-32 and S/Prefix grafts and prunes
- Example

```
host1:boston>show ip dvmrp
Routing Process DVMRP - Distance Vector Multicast Routing Protocol
  Dvmrp Administrative State:         Enabled
  Multicast Administrative State:     Enabled
  Dvmrp Version:                      3.255
  Generation ID:                      0x46828e2b
  Number of Routes:                   2
  Number of Triggered Routes:         0
  Reachable Routes:                   2
  route-hog-notification:             10000
  route-limit:                        7000
  Send-S32-Prunes-Only:               true
  unicastRoutingOnly:                 false
  Graceful Restart Duration:          60
  Graceful Restart is:                complete (timer 0 seconds)
  Redistribution                      None Configured
  dynamic-redistribution:             enabled
```

show ip dvmrp interface

- Use to display DVMRP parameters for the specified interfaces.
- Field descriptions
  - Interface—Type and specifier of the interface connected to a source. For details about interface types and specifiers, see *JUNOSe Command Reference Guide, About This Guide*.
  - SourceAddress—IP address of the interface or, for an unnumbered interface, address of the loopback interface
  - Network/Mask—Network and mask of the subnet on which the interface resides
  - Received Bad Packets/RBdPk—Number of bad packets received on this interface
  - Received Bad Routes/RBdRt—Number of bad routes received on this interface
  - Routes Sent/SntRt—Number of bad routes advertised on this interface
  - Administrative State—Configured state of DVMRP on this interface: enabled or Disabled
  - Summary Address(es)—Specific summary address or addresses that this interface should advertise
  - auto-summary—Status of automatic summarization: Enabled or Disabled
  - metric-offset in—Number of hops by which the router increases a DVMRP route advertised in incoming DVMRP reports
  - metric-offset out—Number of hops by which the router increases a DVMRP route advertised in outgoing DVMRP reports
  - announce-filter—Routes advertised by the interface
  - accept-filter(s)—Names of IP access lists that specify the sources for which the interface accepts routes
- Example 1

```
host1:v3#show ip dvmrp interface
Interface: ATM2/0.1
      SourceAddress:                      1.0.0.1
      Network/Mask:                   1.0.0.1/24
      Received Bad Packets:                    0
      Received Bad Routes:                     0
      Routes Sent:                             0
      Administrative State:              Enabled
      Summary Address(es)          None Configured
      auto-summary:                      Disabed
      metric-offset in:                        1
      metric-offset out:                       0
      announce-filter:                      None
      accept-filter(s)             None Configured
```

- Example 2

```
host1:boston#show ip dvmrp interface brief
  Interface          SourceAddress   Network/Mask        RBdPk RBdRt SntRt
  atm5/0.14          14.0.1.1        14.0.1.1/8             0     0     2
  atm5/0.15          15.0.1.1        15.0.1.1/8             0     0     2
```

show ip dvmrp mroute

- Use to display information about DVMRP routes to multicast groups.
- Field descriptions
  - (S,G) pair—Source, Group pair value
  - Uptime—Time, in seconds, that this (S, G) pair entry has been in the routing table
  - Upstream Prune—Whether the router has sent prune messages for this group
  - RPF Interface—Interface that provides the shortest path back to the source
  - Outgoing interface list—Types and specifiers of interfaces through which the VR forwards DVMRP messages, such as atm3/0. For details about interface types and specifiers, see *JUNOSe Command Reference Guide, About This Guide*.
- Example

```
host1:boston#show ip dvmrp mroute
IP DVMRP Multicast Routing Table
(40.0.0.0/16,  228.1.1.1)  Uptime: 77
  Upstream Prune: none
  RPF Interface
      atm5/0.40
  Outgoing interface list:
      atm5/0.31
```

show ip dvmrp neighbor

- Use to display information about DVMRP neighbors.
- Specify the **brief** keyword to view a summary of information.
- Field descriptions
  - Neighbor Address/NbrAddress—IP address of the neighbor
  - Interface—Interface type and specifier, such as atm3/0. For details about interface types and specifiers, see *JUNOSe Command Reference Guide, About This Guide*.
  - Neighbor upTime/UpTime—Length of time, in seconds, that this router has been a neighbor
  - Neighbor Major Version/Maj—Major number of the DVMRP version on the neighbor
  - Neighbor Minor Version/Min—Minor number of the DVMRP version on the neighbor
  - Neighbor Capabilities/Cap—Capability of the neighbor
    - Prune/P—Ability to send prune messages
    - GenerationId/G—Ability to create a generation ID number
    - Mtrace/M—Ability to trace multicast routes
    - Netmask/N—Ability to send prunes and grafts with a network mask address

- Neighbor State/State—Status of communications with the neighbor
  - Active—Router is able to communicate with this neighbor
  - Down—Neighbor is down
  - Ignoring—Router is not accepting messages from this neighbor
  - Oneway—Router is receiving messages from the neighbor, but the neighbor does not include the router's IP address in the messages. This state can indicate a starting transition, or a problem.
- Generation ID—Number that the neighbor generates each time it boots; when the number changes, the VR discards all information previously learned from the router.
- Routes Received—Number of routes received from this neighbor
- Bad Routes Received—Number of bad routes received from this neighbor
- Bad Packets Received—Number of bad packets received from this neighbor

- Example 1

```
host1:boston#show ip dvmrp neighbor
Neighbor Address:                  14.0.0.1
      Interface:                   atm5/0.14
      Neighbor upTime:               28
      Neighbor Major Version:         3
      Neighbor Minor Version:        255
      Neighbor Capabilities:  Prune  GenerationId  Mtrace  NetMask
      Neighbor State:             Active
      Genereneration ID:        0x3a13fbc2
      Routes Received:                1
      Bad Routes Received             0
      Bad Packets Received:           0
```

- Example 2

```
host1:v3#show ip dvmrp neighbor brief
  Interface          NbrAddress      UpTime Maj Min Cap   State
  atm5/0.14          14.0.0.1            32   3 255 PGMN  Active
  atm5/0.15          15.0.0.1            34   3 255 PGMN  Active
```

show ip dvmrp route

- Use to display information about DVMRP routes.
- Specify an IP address to display the best route to that address.
- Specify an IP address and subnet mask to display the route that exactly matches this IP address and subnet mask
- Specify an interface type and specifier to display routes associated with that interface. For details about interface types and specifiers, see *JUNOSe Command Reference Guide, About This Guide*.
- Specify the **brief** keyword to view a summary of information.

- Field descriptions
  - Prefix—IP address of the network
  - Length—Length of the subnet mask for the network
  - usNbr/Owner—IP address of the upstream neighbor associated with this route or a description of the origin of the route
    - Dvmrp Local—Route is associated with a directly attached network
    - Dvmrp Aggregate—Route is an aggregate route determined by summarization
  - Metric—Metric associated with this interface for this route
  - ExpireTime—Time, in seconds, until the VR starts the process for removing the route
  - UpTime—Length of time, in seconds, that the route has been in the DVMRP routing table
  - Interface—Type and specifier for the interface, such as atm3/0.
- Example 1

```
host1:boston>show ip dvmrp route
Prefix/Length        usNbr/Owner       Metric ExpireTime UpTime Interface
  14.0.0.0/8         Dvmrp Local          1      Never     18   atm5/0.14
    Downstream Interface(s)
        Interface
        atm5/0.15
  15.0.0.0/8         Dvmrp Local          1      Never     18   atm5/0.15
    Downstream Interface(s)
        None
  25.0.0.0/8         14.0.0.1             2        129     11   atm5/0.14
    Downstream Interface(s)
        Interface
        atm5/0.15
```

- Example 2

```
host1:v3#show ip dvmrp route brief
  Prefix/Length      usNbr/Owner       Metric ExpireTime UpTime Interface
  14.0.0.0/8         Dvmrp Local          1      Never     26   atm5/0.14
  15.0.0.0/8         Dvmrp Local          1      Never     26   atm5/0.15
  25.0.0.0/8         14.0.0.1             2        121     19   atm5/0.14
```

show ip dvmrp routeNextHop

- Use to display information about the next hop.
- Field descriptions
  - addr—IP address of the next-hop router
  - mlen—Mask length of the next-hop router
  - ifIndex—SNMP interface index for the interface that connects to the next hop
  - Type—Description of the next-hop router
    - leaf—Neighbor with no downstream neighbors
    - branch—Neighbor with downstream neighbors
- Example

```
host1:boston>show ip dvmrp routeNextHop
  addr/mlen          ifIndex  Type
  172.16.0.0/16            4  leaf
  172.17.0.0/16            4  leaf
  172.18.0.0/16            3  leaf
  172.19.0.0/16            3  leaf
  172.19.0.0/16            4  branch
```

Part 2

Internet Protocol Version 6

Chapter 5

Configuring IPv6 Multicast

IPv6 multicast enables a device to send packets to a group of hosts rather than to a list of individual hosts. This chapter describes how to configure IPv6 multicast on the E-series router; it contains the following sections:

Overview

IPv6 defines three types of addresses: *unicast*, *anycast*, and *multicast*. Each type of address enables a device to send datagrams to selected recipients:

- A unicast address enables a device to send a datagram to a single recipient.
- An anycast address enables a device to send a datagram to one recipient out of a set of recipients.
- A multicast address enables a device to send a datagram to a specified set of hosts, known as a multicast group, in different subnetworks.

IPv6 multicast improves network efficiency by allowing a host to transmit a datagram to a targeted group of receivers. For example, a host may want to send a large video clip to a group of selected recipients. It would be time-consuming for the host to unicast the datagram to each recipient individually. If the host broadcasts the video clip throughout the network, network resources are not available for other tasks. The host uses only the resources it needs when multicasting the datagram.

Routers use multicast routing algorithms to determine the best route and transmit multicast datagrams throughout the network. E-series routers support a number of IPv6 multicast protocols on virtual routers (VRs). Each VR handles the interoperability of IPv6 multicast protocols automatically. To start IPv6 multicast operation on a VR, you access the context for that VR and configure the desired protocols on the selected interfaces. Table 9 describes the function of each the protocol that the router supports.

Table 9: Function of Multicast Protocols on a Router

Protocol	Function
Multicast Listener Discovery (MLD)	Discovers hosts that belong to multicast group.
Protocol Independent Multicast Protocol (PIM)	Discovers other multicast routers that should receive multicast packets.
BGP Multicast Protocol	Routes multicast datagrams between autonomous systems.

The router supports up to 16,384 multicast forwarding entries (multicast routes) at any time.

Reverse-Path Forwarding

IP multicasting uses reverse path forwarding (RPF) to verify that a router receives a multicast packet on the correct incoming interface. The RPF algorithm enables a router to accept a multicast datagram only on the interface from which the router sends a unicast datagram to the source of the multicast datagram.

When the router receives a multicast datagram from a source for a group, the router verifies that the packet was received on the correct RPF interface. If the packet was not received on the correct interface, the router discards the packet. Only packets received on the correct RPF interface are considered for forwarding to downstream receivers.

When operating in sparse-mode, the routers perform an RPF lookup to identify the upstream router from which to request the data and then send join messages for the multicast stream only to that router.

When operating in dense-mode, routers that have multiple paths to the source of the multicast stream initially receive the same stream on more than one interface. In this case, the routers perform an RPF lookup to identify multicast data streams that are not arriving on the best path and send prune messages to terminate these flows.

The RPF lookup need not always be towards the source of the multicast stream. The lookup is done towards the source only when the router is using a source-rooted tree to receive the multicast stream. If the router uses a shared tree instead, the RPF lookup is toward a rendezvous point and not toward the source of the multicast stream.

Multicast Packet Forwarding

Multicast packet forwarding is based on the source (S) of the multicast packet and the destination multicast group address (G). For each (S,G) pair, the router accepts multicast packets on an incoming interface (IIF), which satisfies the RPF check (RPF-IIF). The router drops packets received on IIFs other than the RPF-IIF and notifies the routing protocols that a packet was received on the wrong interface.

The router forwards packets received on the RPF-IIF to a list of outgoing interfaces (OIFs). The list of OIFs is determined by the exchange of routing information and local group membership information. The router maintains mappings of (S,G, IIF) to {OIF1, OIF2…} in the multicast routing table.

You can enable two or more multicast protocols on an IIF. However, only one protocol can forward packets on that IIF. The protocol that forwards packets on an IIF *owns* that IIF. A multicast protocol that owns an IIF also owns the (S,G) entry in the multicast routing table.

Platform Considerations

For information about modules that support IPv6 multicasting on the ERX-7xx models, ERX-14xx models, and the ERX-310 router:

- See *ERX Module Guide, Table 1, Module Combinations* for detailed module specifications.
- See *ERX Module Guide, Appendix A, Module Protocol Support* for information about the modules that support IPv6 multicasting.

For information about modules that support IPv6 multicasting on the E120 router and the E320 router:

- See *E120 and E320 Module Guide, Table 1, Modules and IOAs* for detailed module specifications.
- See *E120 and E320 Module Guide, Appendix A, IOA Protocol Support* for information about the modules that support IPv6 multicasting.

References

For more information about IPv6 multicast, see the following resource:

- A "traceroute" Facility for IP Multicast—draft-ietf-idmr-traceroute-ipm-07.txt (January 2001 expiration)

Before You Begin

You can configure multicast on IPv4 and IPv6 interfaces.

For information about configuring IP interfaces, see *JUNOSe IP, IPv6, and IGP Configuration Guide, Chapter 1, Configuring IP*. For information about configuring IPv6 interfaces, see *JUNOSe IP, IPv6, and IGP Configuration Guide, Chapter 2, Configuring IPv6*.

For information about configuring multicast on IPv4 interfaces, see *Chapter 1, Configuring IPv4 Multicast*.

Configuring the Switching Fabric Bandwidth

By default, the switch fabric for the ERX-1440, ERX-310, E120, and E320 routers uses a bandwidth weighting ratio of 15:2 for multicast-to-unicast weighted round robin (WRR). In the absence of strict-priority traffic, and when both unicast and multicast traffic compete for switch fabric bandwidth, the switch fabric allocates 15/17ths of the available bandwidth to multicast traffic and 2/17ths of the available bandwidth to unicast traffic.

You can use the **fabric weights** command to change the ratio for multicast to unicast traffic on the router switch fabric. For more information about the **fabric weights** command, see *JUNOSe System Basics Configuration Guide, Chapter 5, Managing the System*.

Enabling IPv6 Multicast

In this implementation, IPv6 multicast works on VRs. By default, IPv6 multicast is disabled on a VR. To enable IPv6 multicast on a VR, access the context for a VR, and then issue the **ipv6 multicast-routing** command.

ipv6 multicast-routing

- Use to enable IPv6 multicast routing on the VR.
- By default, IPv6 multicast is disabled on the VR. In the disabled state, all multicast protocols are disabled, and the VR forwards no multicast packets.
- Example

  host1(config)#**ipv6 multicast-routing**

- Use the **no** version to disable IPv6 multicast routing on the VR (the default).

Defining Static Routes for Reverse-Path Forwarding

Use the **ipv6 rpf-route** command to define reverse-path forwarding (RPF) to verify that a router receives a multicast packet on the correct incoming interface.

ipv6 rpf-route

- Use to customize static routes that the router may use for RPF.
- Specify the IPv6 address and subnet mask of the destination network.
- Specify either a next-hop IPv6 address or an interface type and specifier, such as atm 3/0. For details about interface types and specifiers, see *JUNOSe Command Reference Guide, About This Guide*.
- Optionally, specify the distance (number of hops) to the next-hop address.
- Example

  host1(config)#**ipv6 rpf-route 1000::/64 ATM2/1.200**

- Use the **no** version to remove the static route.

Displaying Available Routes for Reverse-Path Forwarding

Use the **show ipv6 rpf-route** command to display all available routes, only the routes to a particular destination, or routes associated with a specific unicast protocol that the router can use for Reverse-Path Forwarding (RPF).

show ipv6 rpf-route

- Use to display routes that the router can use for RPF.
- Specify the IPv6 address and the network mask to view routes to a particular destination.
- Specify the **detail** keyword to view more detailed information about routes to a particular destination.
- Specify a unicast routing protocol to view routes associated with that protocol.

- Field descriptions
  - Protocol/Route type codes—Protocol and route type codes for the table that follows
  - Prefix—Value of the logical AND of the IPv6 address of the destination network and the subnet address
  - Length—Length of the subnet mask in bits
  - Type
    - Connect—Subnet directly connected to the interface
    - Static—Static route
  - Dst—Distance configured for this route
  - Met—Learned or configured cost associated with this route
  - Intf—Type of interface and interface specifier for the next hop. For details about interface types and specifiers, see *JUNOSe Command Reference Guide, About This Guide*.
- Example 1

```
host1#show ipv6 rpf-route
Protocol/Route type codes:
  O- OSPF, E1- external type 1, E2- external type2,
  N1- NSSA external type1, N2- NSSA external type2
  L- MPLS label, V- VR/VRF, *- indirect next-hop

         Prefix/Length              Type    Dst/Met         Intf
-------------------------------- --------- ------- --------------------
11:1:1:10::/60                   Static    1/0     ATM2/0.300
21:2:2:20::/60                   Static    1/0     ATM2/0.300
31:2:2:20::/60                   Connect   0/0     ATM2/0.300
131:1:1:10::/60                  Connect   0/0     ATM2/1.1300
1000::/64                        Static    1/0     ATM2/0.300
```

- Example 2

```
host1#show ipv6 rpf-route 1000::/64 detail

1000::/64 Type:Static  Distance:1  Metric:0
        NextHop:31:2:2:23::2:3  IntfIndex 18  Intf ATM2/0.300
```

Enabling and Disabling RPF Checks

By default, the router accepts multicast packets for each (S,G) pair on an incoming interface (IIF), which satisfies the RPF check (RPF-IIF). When the router performs RPF checks, only the interface that first accepts traffic for an (S,G) pair accepts subsequent traffic for that pair. If traffic stops coming on that interface and starts arriving on another interface, the router does not accept or forward the traffic.

Some network configurations require the router to accept traffic on any interface. To do so, you can disable the RPF check on a specified set of (S,G) pairs by issuing the **ipv6 multicast-routing disable-rpf-check** command.

When you disable RPF checks, the router accepts multicast packets for (S,G) pairs on any incoming interface. When the router has added the new route to its multicast routing table, it accepts multicast packets for these pairs on any interface in the virtual router and forwards them accordingly. Multicast routes established before you issue this command are not affected.

ipv6 multicast-routing disable-rpf-check

- Use to disable RPF checks for specified (S,G) pairs.
- Specify a standard IPv6 access list that defines the (S,G) pairs.
- Example

  host1(config)#**ipv6 multicast-routing disable-rpf-check denver-list**

- Use the **no** version to restore the default situation, in which the router performs RPF checks for all (S,G) pairs.

Using Unicast Routes for RPF

You can use the **ip route-type** command to specify that BGP routes should be available for RPF. Routes available for RPF appear in the multicast view of the routing table.

NOTE: This command functions the same for both IPv4 and IPv6 multicast.

ipv6 route-type

- Use to specify that BGP routes are available only for unicast forwarding, only for multicast RPF checks, or for both.
- Use the **show ipv6 rpf-routes** command to view the routes available for RPF.
- By default, BGP routes are available both for unicast forwarding and multicast reverse-path forwarding checks.
- Example

  host1(config)#**router bgp**
  host1(config-router)#**ipv6 route-type multicast**

- There is no **no** version.

Defining Permanent IPv6 Multicast Forwarding Entries

An mroute is a multicast traffic flow (a (Source, Group) entry used for forwarding multicast traffic). By default, forwarding mroutes (with a valid RPF incoming interface) are timed out if data for them is not received for 210 seconds. However, you can specify an mroute as permanent by using the **ipv6 multicast-routing permanent-mroute** command.

ipv6 multicast-routing permanent-mroute

- Use to specify that any newly created mroutes that match the specified access-list do not time out.
- Using this command does not change existing mroutes.
- Permanent mroutes are removed if a topology change occurs that affects the mroute.
- Permanent mroutes may be removed due to certain protocol actions (for example, PIM sparse mode switching from shared to shortest path tree).
- Outgoing interface lists of permanent mroutes may change due to protocol actions.
- Example

  host1(config)#**ipv6 multicast-routing permanent-mroute routesv61**

- Use the **no** version to prevent any new mroutes from becoming permanent. To remove existing permanent mroutes, use the **clear ipv6 mroute** command.

Defining a Multicast Bandwidth Map

Multicast interface-level admission control, port-level admission control, and QoS adjustment all use a single multicast bandwidth map. The multicast bandwidth map is a route map that uses the **set admission-bandwidth**, **set qos-bandwidth**, **set admission-bandwidth adaptive**, or **set qos-bandwidth adaptive** commands. The **adaptive** commands configure an auto-sense mechanism for measuring the multicast bandwidth.

NOTE: Even though you can include any of the above commands several times in a route map entry, only the last admission-bandwidth command or qos-bandwidth command in the bandwidth map is used. In other words, if you included the **set qos-bandwidth** command first and then the **set qos-bandwidth adaptive** command, the bandwidth map would use the **set qos-bandwidth adaptive** command.

Interface- and port-level admission control is performed when an OIF on the interface or port is added to the mroute for a given (S,G) multicast data stream and the multicast bandwidth map contains a **set admission-bandwidth** or **set admission-bandwidth adaptive** action for that (S,G).

QoS adjustment is performed on the joining interface when an OIF is added to the mroute for a given (S,G) data stream and the multicast bandwidth map contains a **set qos-bandwidth** or **set qos-bandwidth adaptive** action for that (S,G).

NOTE: You can create a single route map with the **set admission-bandwidth** command, the **set qos-bandwidth** command, or both. However, creating an entry with only one of these **set** commands enables only that specific function for the matched address (that is, only multicast traffic admission control or only QoS adjustment). The same is true for the **adaptive** commands.

Using the Auto-Sense Mechanism

Video bandwidth is typically considered to be a constant rate—2 Mbps for standard definition television (SDTV) and 10 Mbps for high definition television (HDTV). However, in reality, and depending on achievable video compression, the bit rate can vary. For example, HDTV streams (using MPEG4 or WM9 encoding) can vary between 6 Mbps (for low-action programs) to 10 Mbps (for a fast-paced, high-action programs). The auto-sense mechanism allows the bandwidth value, used for admission control and QoS adjustment, to be the actual measured rate of the stream. Using this feature to measure the actual bandwidth avoids the need to configure arbitrary bandwidth limits and enables a channel to be reassigned to a different (S, G) without requiring a bandwidth map to be changed.

How Adaptive Mode Works

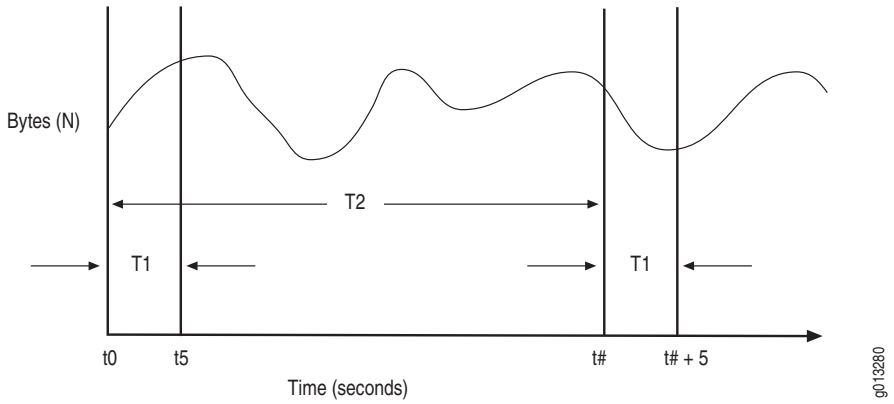
As mentioned above, you configure the auto-sense mechanism in the multicast bandwidth using the **set admission-bandwidth adaptive** command, **set qos-bandwidth adaptive** command, or both. For example:

```
host1(config)#route-map mcast-bandwidths permit 10
host1(config-route-map)#match ip address sdtv
host1(config-route-map)#set admission-bandwidth adaptive
host1(config-route-map)#set qos-bandwidth adaptive
host1(config-route-map)#end
```

In this example, any stream with an (S,G) that matches the sdtv access list performs adaptive bandwidth detection for admission control and QoS adjustment.

A rate measurement mechanism runs on the ingress line card that polls the forwarding controller (FC) to obtain statistics for each mroute. This mechanism then reports the rate measurement to the SRP to update the bandwidth map. By computing the average bandwidth over a relatively short sampling period (T1; 5 seconds), the measurement approximates the peak bandwidth of the multicast stream.

As an example, assume that a new mroute (S1, G1) is added to the interface controller (IC) at time t0.



To calculate the measured bandwidth of a stream, the router uses the following equation:

$R = (N_{t+5} - N_t) / 5$

Where

R = Calculated bandwidth of the stream during each sampling interval

N_t = Bytes measured at each determined time interval (t seconds)

N_{t+5} = Bytes measured 5 seconds after each determined time interval (t seconds)

NOTE: When the mroute is first installed in the FC (at t = 0), R_0 is undetermined. For multicast admission control no joins are admitted until the first bandwidth measurement is computed (that is, for admission control, R_0 is considered to be infinite). Similarly, no Qos adjustment occurs until the first bandwidth measurement is computed (that is, for Qos adjustment, R_0 is considered to be zero [0]).

Using the earlier graph as a reference, the first bandwidth rate (R_1) is determined by calculating the number of bytes received during the first sampling period, T_1. Mroute statistics are read at time t_0 (N_0) and at time t_5 (N_5) and the bytes received values are subtracted and divided by the time period T_1 to yield the average rate. This process is repeated every sampling interval, T_2, to yield rates R1, R2, R3, and so on.

The first two sampling interval calculations would look like the following:

$R_1 = (N_5 - N_0)/5$

$R_2 = (N_{\#+5} - N_\#)/5$

The router maintains a history of bandwidth measurements (H) for each mroute, up to a maximum of M measurements. The actual rate, R, reported to the SRP is the maximum rate measured in those H samples.

In order to minimize the IC to SRP traffic generated by the rate measurements, the IC reports a bandwidth change only when a newly computed rate (R#) differs from the current rate by a specified threshold. When R_s is computed at time t = 5 seconds, R is set to R_1. A rate update occurs whenever a newly calculated rate (R) differs from R_1 by at least a threshold value (specified as a percentage, P) of the measured peak bandwidth. This calculation would look like the following:

R = Rt, if and only if the absolute value of (R - Rt) > P * R.

The values assigned to variables associated with this algorithm are as shown in Table 10.

Table 10: Adaptive Mode Algorithm Values

Variable	Value	Units	Description
T1	5	Seconds	Sampling period; the time in which a sample is taken
T2	0	Seconds	Sampling interval; zero (0) seconds indicates continuous sampling
H	12	Samples	Number of history samples over which to compute measurement
M	12	Samples	Maximum number of samples maintained in history
P	1	Percent	Threshold value; percent difference by which a newly calculated rate must differ from the measured peak bandwidth before a rate update occurs

Multicast Bandwidth Map Example

The following example creates a multicast bandwidth map for both multicast traffic admission control and QoS adjustment:

NOTE: In this example, you can replace the **set admission-bandwidth** command and **set qos-bandwidth** command with their **adaptive** command counterparts.

1. Define a route-map using the **set admission-bandwidth** and **set qos-bandwidth** commands.

```
host1(config)#route-map mcast-bandwidths permit 10
host1(config-route-map)#match ipv6 address sdtv
host1(config-route-map)#set admission-bandwidth 2000000
host1(config-route-map)#set qos-bandwidth 2000000
host1(config-route-map)#route-map mcast-bandwidths permit 20
host1(config-route-map)#match ipv6 address hdtv
host1(config-route-map)#set admission-bandwidth 10000000
host1(config-route-map)#set qos-bandwidth 10000000
host1(config-route-map)#end
```

2. Define the access list for use by the **match ipv6 address** command to match (S,G) and (*,G) entries.

```
host1(config)#access-list sdtv permit ip host 31::1 ff3e::0/112
host1(config)#access-list hdtv permit ip host 32::1 ff3e::0/112
host1(config)#access-list hdtv permit ip host 32::2 ff3e::0/112
host1(config-route-map)#end
```

NOTE: You can also define a prefix-list or a prefix-tree for use by the **match ipv6 address** command to match (S,G) and (*,G) entries.

For additional information about configuring QoS adjustment, see *Configuring Multicast QoS Adjustment* on page 148.

For additional information about configuring interface- and port-level admission control, see *Blocking and Limiting Multicast Traffic* on page 159.

For additional information about creating route maps, see *JUNOSe IP Services Configuration Guide, Chapter 1, Configuring Routing Policy*.

Configuring Multicast QoS Adjustment

When the router uses multicast OIF mapping, any multicast streams that a subscriber receives bypass any configured QoS treatment for that subscriber interface. The Multicast QoS adjust feature provides a way in which the router can account for this multicast traffic.

NOTE: For additional information about how to configure OIF mapping, see *Configuring Group Outgoing Interface Mapping* on page 53.

The following sections provide two possible configuration cases for using multicast QoS adjustment.

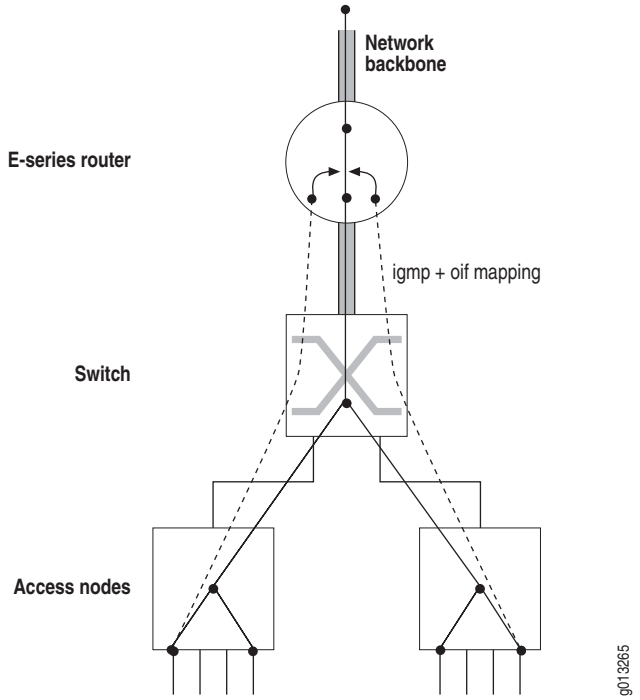
NOTE: For additional information about QoS adjustment, see *JUNOSe Quality of Service Configuration Guide, Chapter 24, Configuring a QoS Parameter*.

Multicast OIF Mapping Case

Multicast OIF mapping enables the router to decrease the inefficiencies associated with replicating streams of multicast traffic. Using OIF maps, MLD joins that the router receives on a subscriber interface can be mapped to a special interface for forwarding. This special interface can be on a different physical port or line module from that of the join interface.

Using this mapping function, the router can send a single copy of each multicast stream over the special interface and the access nodes are configured to perform any final replication to the subscribers and merge unicast and multicast data flows onto the subscriber interfaces as necessary. See Figure 13.

Figure 13: Multicast OIF Mapping



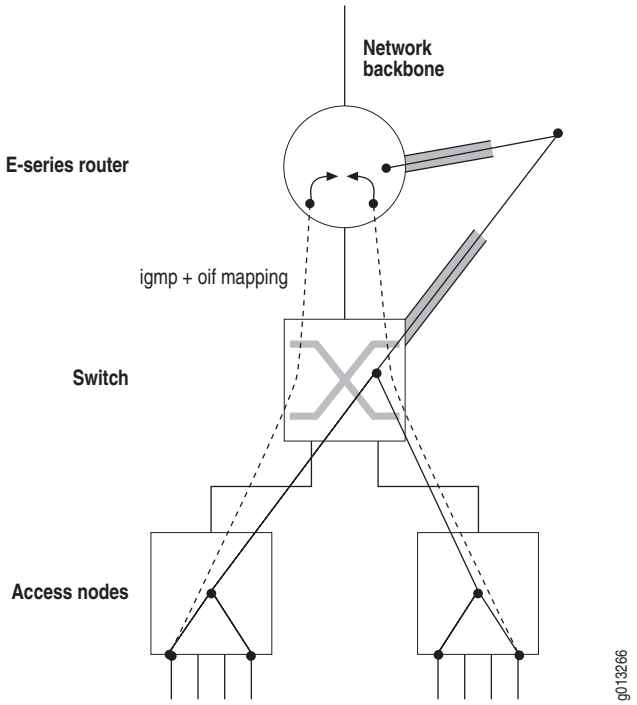
One disadvantage to using multicast OIF mapping is that the multicast traffic bypasses any QoS treatment that is applied to subscriber interfaces. Configuring QoS adjustment resolves this problem. (See *JUNOSe Quality of Service Configuration Guide, Chapter 24, Configuring a QoS Parameter* for additional information about configuring QoS adjustment.) With QoS adjustment configured, when a subscriber requests to receive a multicast stream (or, more appropriately, when an OIF is added to the mroute), the router reduces the unicast QoS bandwidth applied to the subscriber interface (that is, the join interface) by the amount of bandwidth for that multicast stream.

Multicast Traffic Receipt Without Forwarding

In this case, the router is not given the responsibility of forwarding multicast streams. Instead, the service provider arranges for the router to receive the multicast streams so the router can detect the flow and perform QoS adjustment. An OIF map is installed that maps the traffic streams to a loopback interface configured for MLD version passive. This means that when the traffic is received, a null mroute is installed (that is, an mroute with an empty OIF list) and the router applies the QoS adjustment to the join interface. See Figure 14.

NOTE: Ensure that PIM-SM (or any other upstream multicast protocol) is informed of the group (or source-group) interest.

Figure 14: Multicast Traffic Receipt Without Forwarding



Activating Multicast QoS Adjustment Functions

The **ipv6 multicast-routing bandwidth-map** command activates the specified bandwidth map. By activating the bandwidth map, this command also activates the multicast QoS adjustment function contained in the bandwidth map.

CAUTION: To activate multicast QoS adjustment, you must first create a bandwidth map. See *Defining a Multicast Bandwidth Map* on page 144 for details.

ipv6 multicast-routing bandwidth-map

- Use to enable the QoS adjust function on the router.
- Example

  host1(config)#**ipv6 multicast-routing bandwidth-map mcast-bandwidths**

- Use the **no** version to disable the multicast QoS adjustment function on the router.

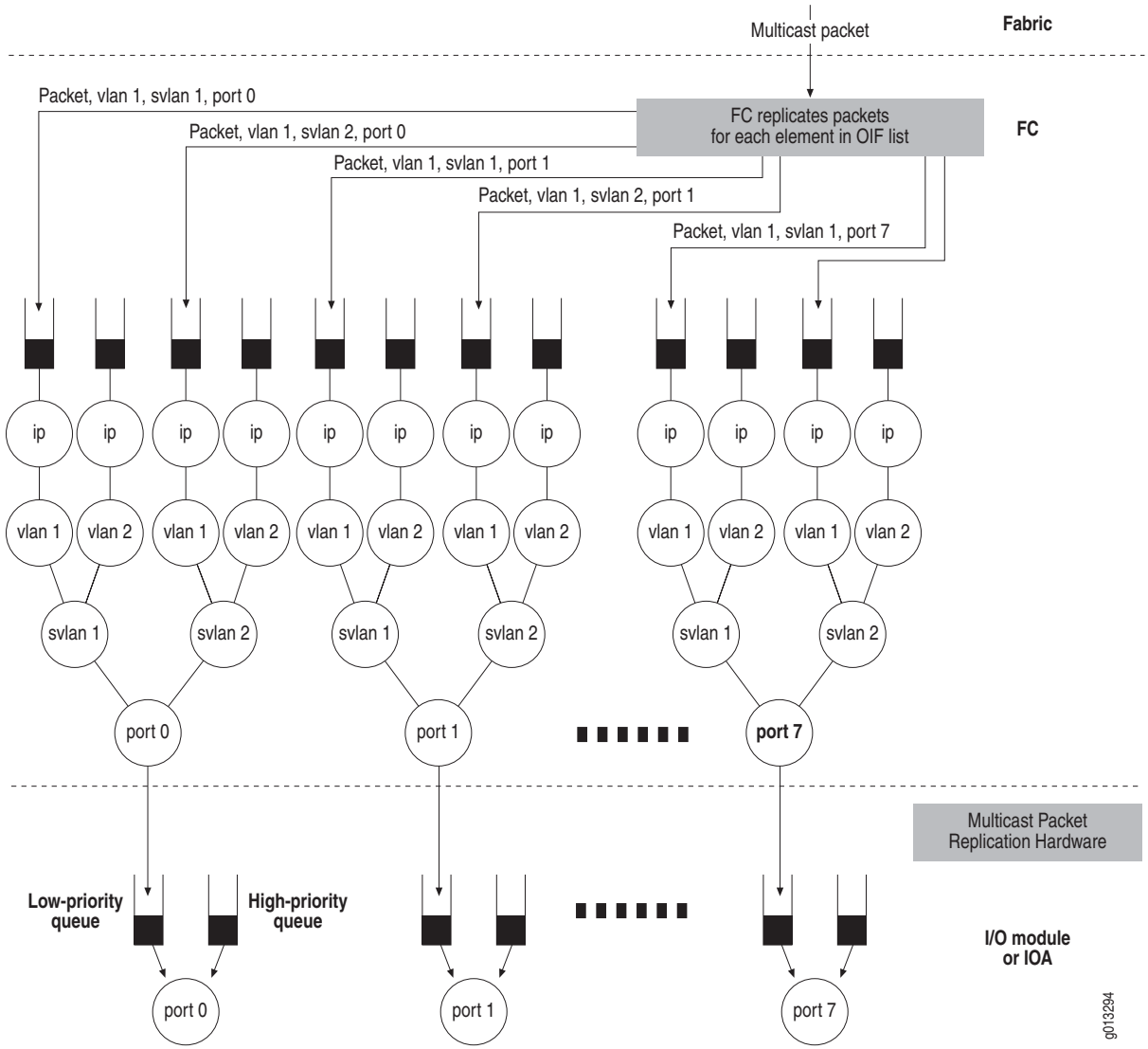
Configuring Hardware Multicast Packet Replication

You can configure IPv6 multicast to replicate packets to optimized hardware on a logical port instead of using the forwarding controller (FC) on the router.

The bandwidth between the line module and the I/O module or IOA on the E-series router is limited. A high-density Ethernet module provides eight physical ports that can consume the bandwidth between the line module and the I/O module or IOA before providing enough traffic to support egress line rate for all of these ports.

Figure 15 displays how multicast traffic is typically replicated on the line module. Each of these replicated packets is transmitted from the line module to the I/O module or IOA.

Figure 15: Packet Flow Without Hardware Multicast Packet Replication



The hardware multicast packet replication feature enables you to configure multicast traffic for a VLAN or S-VLAN to be replicated on the I/O module or IOA so that only one copy of the packet is transmitted from the line module to the I/O module or IOA. Replication for each of the ports is performed on the I/O module or IOA.

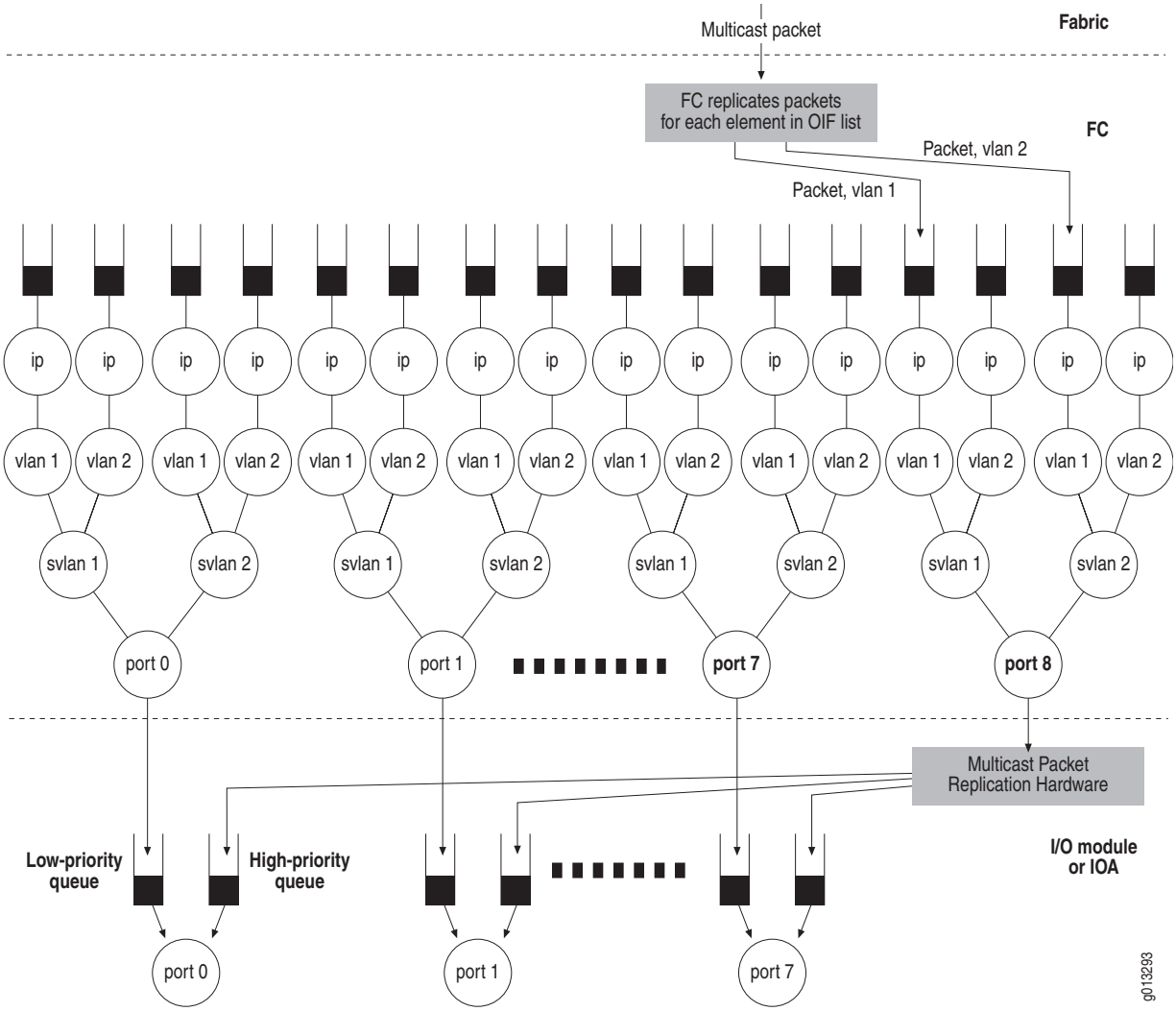
Configuring hardware multicast packet replication for high-density Ethernet is useful when you want to provide the same multicast stream out of some or all of the ports, such as for IP television (IPTV). Configuring hardware multicast packet replication enables you to:

- Reduce the number of packets sent from the FC to the module.
- Reduce the CPU consumed by the FC processing each elaboration of the packet.

You can use the feature to increase the bandwidth of multicast traffic out of each of the Gigabit Ethernet ports.

Figure 16 on page 153 displays the flow of a multicast packet using the hardware multicast packet feature.

Figure 16: Packet Flow with Optimized Multicast Packet Replication



Each high-density Ethernet module has eight physical ports, numbered 0–7. A logical port is available for the hardware multicast packet replication feature, numbered port 8.

JUNOSe tracks the OIFs in an mroute that have been redirected to use the hardware multicast packet replication hardware. The system accepts only egress multicast traffic to traverse the interface stack on the enabled port. The system drops unicast traffic that is routed to this port.

Each port on the I/O module or IOA displayed in Figure 16 has two queues. These queues are further down the egress path than the queues found on the line module and populated by the FC.

The low-priority queue is dedicated to packets that are received from the line module queues that are dedicated to the physical ports. This queue blocks when full and provides backpressure to the line module. This queue services unicast and multicast traffic that is not using the hardware multicast packet replication feature.

The high-priority queue is dedicated to packets that are received from the line module queue for port 8. This queue is serviced at a higher priority than the first queue, and drops packets when full.

For more information about high-density Ethernet, see *JUNOSe Physical Layer Configuration Guide, Chapter 5, Configuring Ethernet Interfaces*.

Supported Modules and Encapsulations

You can enable optimized multicast packet replication on port 8 of the following high-density Ethernet modules:

- GE-8 I/O module (pairs with the GE-HDE line module)
- ES2-S1 GE-8 IOA (pairs with the ES2 4G LM and the ES2 10G LM)

When enabled, the optimized multicast packet replication feature defines the encapsulation of the egress multicast packet. The following encapsulations are supported:

- IPv6 over Gigabit Ethernet
- IPv6 over VLAN
- IPv6 over S-VLAN

NOTE: 802.3ad link aggregation group (LAG) bundles do not support optimized multicast packet replication.

The optimized multicast packet replication feature also provides an interface over which you can configure the following:

- IP MTU
- Ethernet MTU
- Egress IP policy
- Egress VLAN policy
- QoS

Relationship with OIF Mapping

Multicast OIF mapping enables the router to decrease the inefficiencies associated with replicating streams of multicast traffic. Using OIF maps, MLD joins that the router receives on a subscriber interface can be mapped to a special interface for forwarding.

The hardware multicast packet replication feature enables you to redirect each of the IPv6 interfaces on a line module over a dedicated multicast VLAN to a single IPv6 interface over port 8. The FC is only required to send a single packet per dedicated multicast VLAN to the I/O module or IOA. The module then replicates this packet to the appropriate ports.

For more information about configuring OIF mapping, see *Configuring Group Outgoing Interface Mapping* in *Chapter 6, Configuring Multicast Listener Discovery*.

Hardware Multicast Packet Replication Considerations

When configuring hardware multicast packet replication, the following considerations apply.

- Do not configure or transmit routing protocols over port 8. The FC drops traffic routed to an IPv6 interface stacked over port 8.
- We recommend that you configure the IP address of the IPv6 interface over port 8 to be unnumbered.
- We recommend that you configure an IPv6 interface over a VLAN over one of the physical ports to reference the IPv6 interface over the same VLAN over port 8.

  You cannot create the following configurations:

  - When two IPv6 interfaces configured over a port reference the same IPv6 interface over port 8. The system does not accept this configuration attempt because you typically configure the hardware multicast packet replication feature to redirect multicast traffic over one VLAN, then redirect it to the same VLAN on port 8.
  - When the IPv6 interface configured with the hardware multicast packet replication attribute is not installed on a line module that supports hardware multicast packet replication.
  - When the IPv6 interface designated by the hardware multicast packet replication attribute is not installed on a line module that supports hardware multicast packet replication.
  - When the IPv6 interface designated by the hardware multicast packet replication attribute is not on the same line module as the IPv6 interface configured with this attribute.

- When you configure a unique source MAC address for VLANs on port 8, the hardware multicast packet replication hardware stamps the source MAC address on the VLAN, overwriting any MAC address that you configured. For more information, see *JUNOSe Physical Layer Configuration Guide, Chapter 5, Configuring Ethernet Interfaces*.

- The regular multicast implementation utilizes interface stacking that provides a unique IPv6 attachment point for each elaboration of the egress multicast packet.

  For the hardware multicast packet replication feature, you must attach policies to an interface stack over port 8 that defines the encapsulation of the egress multicast traffic. The system supports policies over port 8 just as it is above any of the other ports on this line module.

  Policies applied to the interface stack over port 8 affect the packets traversing this stack whether or not the packet is destined for one port or all of the physical ports. Therefore, you cannot apply different egress policies to multicast traffic for the interfaces stacked above different ports, or rate limit on an individual interface over a port. You also cannot monitor policy statistics on individual interfaces over a port.

  Instead, you can apply egress policy to an interface stacked over port 8. The system applies the policy before the packet has been elaborated for each of the ports.

- The JUNOSe QoS component provides hierarchical egress scheduling and shaping on Gigabit Ethernet ports 0–7. The regular multicast implementation replicates packets on the FC, with each replicated packet placed on a line module queue destined for a single physical port. The line module queue can also receive QoS behavior specific to that queue.

  For the hardware multicast packet replication feature, the FC does not replicate the packet for each of the individual ports. Instead, it places the packet on a special queue destined for port 8.

  You can configure QoS on the packets flowing through port 8, but this has limited value because each packet passed through this port can be transmitted through one of more of the physical ports. Therefore, the packets placed on this special queue might not receive the same QoS behavior as ports 0–7.

  We recommend that you configure the network so the I/O or IOA queues are not oversubscribed. The traffic transmitted by the physical port is a combination of packets from the two I/O or IOA queues. When the sum of the packets in these queues is greater than line rate, the system can drop traffic that is not using hardware multicast packet replication.

  When you configure a traffic shaper on a physical port and configure hardware multicast packet replication, the packets created using the feature avoid the traffic shaper for that port. To control this, you can use traffic shaper on the physical port and port 8. The sum of the traffic shapers must be less than or equal to the line rate of the port.

A traffic shaper on port 8 can result in the overall utilization of egress bandwidth for any one port being less the line rate because the packets being replicated might not be transmitted to every port. Packets destined to some of the ports contribute to the traffic shaping for all of the ports on the I/O module or IOA.

Configuring Hardware Multicast Packet Replication

To configure hardware multicast packet replication:

1. Configure port 8 on a high-density Ethernet module to accept redirected egress multicast traffic.

   a. Specify the Gigabit Ethernet interface on port 8.

   b. Create a VLAN major interface.

   c. Create a VLAN subinterface.

   d. Assign a VLAN ID.

   e. Configure an unnumbered IPv6 interface.

   f. Enable MLD on the interface with only multicast-data-forwarding capability.

   ```
   host1(config)#interface gigabitEthernet 2/8
   host1(config-if)#encapsulation vlan
   host1(config-if)#interface gigabitEthernet 2/8.1
   host1(config-if)#vlan id 1
   host1(config-if)#ipv6 unnumbered loopback 0
   host1(config-if)#ipv6 mld version passive
   ```

2. Configure an IPv6 interface to redirect egress multicast traffic to port 8.

   a. Create a VLAN subinterface.

   b. Assign a VLAN ID.

   c. Assign an IPv6 address.

   d. Configure the interface to redirect egress multicast traffic to port 8.

   ```
   host1(config)#interface gigabitEthernet 2/0.101
   host1(config-if)#vlan id 1
   host1(config-if)#ipv6 address 1::1/64
   host1(config-if)#ipv6 multicast ioa-packet-replication gigabitEthernet 2/8.1
   ```

encapsulation vlan

- Use to configure VLAN as the encapsulation method for the interface.
- Example

  ```
  host1(config-if)#encapsulation vlan
  ```

- Use the **no** version to disable VLAN on an interface.

ipv6 mld version

- Use to set the MLD version (1 or 2) for the interface.
- Example

  host1:boston(config-if)#**ipv6 mld version 2**

- Use the **no** version to set the version to the default, MLDv2.

ipv6 multicast ioa-packet-replication

- Use to configure hardware multicast packet replication on port 8 of a high-density Ethernet module.
- Example

  host1(config-if)#**ipv6 multicast ioa-packet-replication gigabitEthernet 3/8.1**

- Use the **no** version to disable hardware multicast packet replication.

ipv6 unnumbered

- Use to configure an unnumbered IPv6 interface.
- This command enables IPv6 processing on an interface without assigning an explicit IPv6 address to the interface.
- You must specify an interface location, which is the identifier of another interface on which the router has an assigned IPv6 address. This interface cannot be another unnumbered interface.
- Example

  host1(config-if)#**ipv6 unnumbered loopback 10**

- Use the **no** version to disable IPv6 processing on the interface.

Monitoring Optimized Multicast Packet Replication

This section describes how to monitor hardware multicast packet replication.

Port Statistics

Use the **show interfaces gigabitEthernet** command to display port statistics for port 8. For port 8, queue statistics have no direct relationship to any of the 8 ports because each packet transmitting through the queue can be sent through 1 or more of the 8 physical ports. For more information, see *Monitoring Ethernet Interfaces* in *JUNOSe Physical Layer Configuration Guide, Chapter 5, Configuring Ethernet Interfaces*.

IP and VLAN Statistics

Use the **show vlan subinterface** command to display statistics for a VLAN interface configured over port 8. For more information, see *Monitoring Ethernet Interfaces* in *JUNOSe Physical Layer Configuration Guide, Chapter 5, Configuring Ethernet Interfaces*.

Use the **show ipv6 interface** command to display statistics for an IPv6 interface configured over port 8. For more information, see *Monitoring IPv6* in *JUNOSe IP, IPv6, and IGP Configuration Guide, Chapter 2, Configuring IPv6*.

Multicast traffic redirected by the hardware multicast packet replication feature is displayed in the statistics for the IPv6 or VLAN interface over port 8, not the original IP or VLAN interface over the physical port.

The statistics for the IPv6 or VLAN interface over port 8 reflect the number of packets that passed through this interface destined for the hardware multicast packet replication hardware. These statistics have no direct correlation to the number of packets being transmitted from any of the physical ports.

MLD Statistics

Use the **show ipv6 mld interface** command to display statistics, including hardware multicast packet replication status, for an IPv6 interface stacked over port 8. For more information, see *Monitoring MLD* in *Chapter 6, Configuring Multicast Listener Discovery*.

Blocking and Limiting Multicast Traffic

You can either block mroute creation, limit the multicast bandwidth admitted on an outgoing interface, or limit outgoing interface creation on a port.

Blocking Mroutes

By default, when an interface receives multicast traffic, even when the scope of that traffic exceeds link-local, the virtual router creates an mroute. You can use the **ipv6 block-multicast-sources** command to block all multicast traffic with a scope larger than link-local (for example, global) and prevent mroute creation under these conditions.

NOTE: Issuing this command does not affect reception of link-local multicast packets.

ipv6 block-multicast-sources

- Use to prevent mroute creation by blocking multicast traffic that has a scope larger than link-local (for example, global).
- Example

  host1(config)#**ipv6 block-multicast-sources**

- Use the **no** version to restore the default behavior of creating mroutes on receiving multicast packets.

Limiting Interface Admission Bandwidth

Interface-level multicast admission control is performed when an OIF on the interface is added to the mroute for a given (S,G) multicast data stream and the multicast bandwidth map contains a **set admission-bandwidth** action for that (S,G).

When enabled, the admission-bandwidth for a particular (S,G) is read from the multicast bandwidth map and recorded in the mroute when the (S,G) mroute is created. When an IOF is subsequently added to the mroute, the OIF is blocked from forwarding data if the additional bandwidth contributed by the (S,G) would exceed the admission-bandwidth limit for the interface.

CAUTION: Before you can limit interface-level admission bandwidth, you must first create a bandwidth map. See *Defining a Multicast Bandwidth Map* on page 144 for details.

Enabling Interface Admission Bandwidth Limitation

You can use the **ipv6 multicast admission-bandwidth-limit** command to enable multicast admission control on interfaces (including dynamic IP interfaces) that are configured to run MLD. You can also use this command on a PIM (sparse-mode, dense-mode, or sparse-dense-mode) interface if MLD is configured on the interface (including the **ipv6 mld version passive** command).

ipv6 multicast admission-bandwidth-limit

- Use to limit bandwidth for an interface that accepts MLD groups.
- Use on any interface configured to run MLD.
- Can also configure on a PIM (sparse-mode, dense-mode, or sparse-dense-mode) interface if MLD (which you can configure using the **ipv6 mld version passive** command) is also configured on the interface.
- Example

  host1:boston(config-if)#**ipv6 multicast admission-bandwidth-limit 2000000**

- Use the **no** version to remove the bandwidth limitation for the interface.

OIF Interface Reevaluation Example

If you change the admission bandwidth for an interface, all mroutes with that interface as an OIF are reevaluated as follows:

- If the bandwidth limit is increased, blocked OIFs may become unblocked. If the interface is a blocked OIF on multiple mroutes, the order in which the mroutes are visited, and which (S,G) streams become unblocked, is not specified.
- If the bandwidth limit is decreased, no currently admitted OIFs are blocked. However, no new OIFs are admitted until the total admitted bandwidth for the interface drops below the new limit.
- If the bandwidth is increased to the point that the bandwidth limit for an interface is now exceeded, no currently admitted OIFs for the affected mroutes are blocked. However, no new OIFs are admitted until the total admitted bandwidth drops below the configured limit.

NOTE: If the multicast bandwidth map that includes the **set admission-bandwidth command** is changed, all affected mroutes are reevaluated in the same manner described previously.

As an example of this function, if the interface has accepted a total bandwidth of 2000000 bps, and you set a limit of 1000000 bps on the interface, the router does not disconnect any already connected OIFs but prevents the interfaces from accepting any more groups. Over time, some groups leave the interfaces and, eventually, the interface limit of 1000000 bps is reached and maintained by the router.

If you set limits for both a port and interfaces on that port, the router uses the lower of the two limits when determining whether or not an interface can accept any new MLD groups. For example, if you specify an admission bandwidth limit of 2000000 bps for the port and 3000000 bps groups for each interface, additional groups can only be accepted until the port limit of 2000000 bps is reached.

Creating Mroute Port Limits

When a multicast forwarding entry (that is, an mroute) is added with an outgoing interface (OIF) on a port, the OIF count for that port is incremented. If you configure a port limit and the OIF count on the port count exceeds that limit, no OIFs on that port are added to mroutes (that is, new OIFs are blocked).

mroute port limit

- Use to configure a limit on the number of mroute OIFs that can be added across different virtual routers, on a port.
- Example

  host1(config)#**mroute port 3/0 limit 10**

- Use the **no** version to remove any OIF port limits.

Limiting Port Admission Bandwidth

Port-level multicast admission control is performed when an OIF on that port is added to the mroute for a given (S,G) multicast data stream and the multicast bandwidth map contains a **set admission-bandwidth** action for that (S,G).

When enabled, the admission-bandwidth for a particular (S,G) is read from the multicast bandwidth map and recorded in the mroute when the (S,G) mroute is created. When an IOF is subsequently added to the mroute, the OIF is blocked from forwarding data if the additional bandwidth contributed by the (S,G) would exceed the admission-bandwidth limit for the port on which the interface resides.

CAUTION: Before you can limit port-level admission bandwidth, you must first create a bandwidth map. See *Defining a Multicast Bandwidth Map* on page 144 for details.

Enabling Port Admission Bandwidth Control

You can use the **mroute port admission-bandwidth-limit** command to limit the total multicast bandwidth that can be admitted on a port. The admitted bandwidth is summed across all virtual routers with IPv4 and IPv6 mroutes that have OIFs on the port.

NOTE: Admission bandwidth values for a given (S,G) mroute are determined from the bandwidth map. See *Defining a Multicast Bandwidth Map* on page 144 for details.

mroute port admission-bandwidth-limit

- Use to configure a limit on the admission bandwidth of OIFs containing IPv4 or IPv6 mroutes, across different virtual routers, on a port.
- Example

  host1(config)#**mroute port admission-bandwidth-limit 3000000**

- Use the **no** version to remove any OIF admission bandwidth limits.

OIF Port Reevaluation Example

If you change the admission bandwidth for a port, all mroutes with an OIF on that port are reevaluated as follows:

- If the bandwidth limit is increased, blocked OIFs can become unblocked. However, the order in which the mroutes are visited, and which (S,G) streams become unblocked, is not specified.
- If the bandwidth limit of a port is decreased, no currently admitted OIFs are blocked. However, no new OIFs are admitted until the total admitted bandwidth for the port drops below the new limit.
- If the bandwidth is increased to the point that the bandwidth limit for an interface is now exceeded, no currently admitted OIFs for the affected mroutes are blocked. However, no new OIFs are admitted until the total admitted bandwidth drops below the configured limit.

NOTE: If the multicast bandwidth map that includes the **set admission-bandwidth command** is changed, all affected mroutes are reevaluated in the same manner described previously.

As an example of this function, if the port has accepted a total bandwidth of 3000000 bps, and you set a limit of 2000000 bps on the port, the router does not disconnect any already connected OIFs but prevents the interfaces from accepting any more groups. Over time, some groups leave the interfaces and, eventually, the port limit of 2000000 bps is reached and maintained by the router.

If you set limits for both a port and interfaces on that port, the router uses the lower of the two limits when determining whether or not an interface can accept any new MLD groups. For example, if you specify an admission bandwidth limit of 2000000 bps for the port and 3000000 bps groups for each interface, additional groups can only be accepted until the port limit of 2000000 bps is reached.

Deleting Multicast Forwarding Entries

You can clear one or more forwarding entries from the multicast routing table. However, if you do so, the entries may reappear in the routing table if they are rediscovered.

clear ipv6 mroute

- Use to delete IPv6 multicast forwarding entries.
- If you specify an *, the router clears all IP multicast forwarding entries.
- If you specify the IPv6 address of a multicast group, the router clears all multicast forward entries for that group.
- If you specify the IPv6 address of a multicast group and the IPv6 address of a multicast source, the router clears the multicast entry that matches that group and source.
- Example

  host1:boston#**clear ipv6 mroute ***

- There is no **no** version.

Monitoring IPv6 Multicast Settings

The commands in this sections display general information about the IPv6 multicast configuration on the router.

show ipv6 mroute

- Use to display information about all or specified multicast forwarding entries.
- Specify a multicast group IPv6 address or both a multicast group IPv6 address and a multicast source IPv6 address to display information about particular multicast forwarding entries.
- Use the **summary** option to see a summary rather than a detailed description.
- Use the **count** option to display the number of multicast forwarding entries.
- Use the **statistics** option to display statistics for packets received through all multicast forwarding entries that the router has added to the multicast routing table and established on the appropriate line modules.
- Use the **active** option to display the active multicast routes with admission bandwidth greater than the specified bandwidth threshold. The default is 4000 bps.
- Field descriptions
  - (S,G)—IPv6 addresses of the multicast source and the multicast group
  - Admission bandwidth—Admission bandwidth (in bps)
  - QoS bandwidth—QoS bandwidth (in bps)
  - Uptime—Length of time that the (S,G) pair has been active, in *days hours:minutes:seconds* format

- Expires—Length of time for which the (S,G) pair will be active, in *days hours:minutes:seconds* format
- RPF Route—IPv6 address and prefix of the RPF route
- Incoming interface—Type and specifier of the incoming interface for the RPF route
- neighbor address—IPv6 address of the neighbor
- owner—Owner of the route
  - Local—route belonging to the local interface
  - Static—Static route
  - Other protocols—Route established by a protocol
- Incoming interface list—List of incoming interfaces on the router. Details include:
  - Type of interface and its specifier
  - Action that the interface takes with packets: accept or discard
  - Multicast protocol that owns the interface
  - Time that the interface has been active in this multicast forwarding entry, in *days hours:minutes:seconds* format
  - Time that the interface will cease to be active in this multicast forwarding entry, in *days hours:minutes:seconds* format
- Outgoing interface list—List of outgoing interfaces on the router. Details include:
  - Type of interface and its specifier
  - Action that the interface takes with packets: forward
  - Protocol running on the interface: PIM or MLD
  - Time that the interface has been active in this multicast forwarding entry, in *days hours:minutes:seconds* format
  - Time that the interface will cease to be active in this multicast forwarding entry, in *days hours:minutes:seconds* format
- Counts—Numbers of types of source group mappings
  - (S,G)—Number of (S,G) entries
  - (*,G)—Number of (*,G) entries

- Example

```
host1#show ipv6 mroute
IP Multicast Routing Table

(S, G) uptime d h:m:s[, expires d h:m:s]
  [Admission bandwidth: bps]
  [QoS bandwidth: bps]
  RPF route: addr/mask, incoming interface
             neighbor address, owner route-owner
  Incoming interface list:
      Interface (addr/mask), State/Owner [(RPF IIF)]
  Outgoing interface list:
      Interface (addr/mask), State/Owner, Uptime/Expires
```

```
(10:0:0:1:1::, ff0e::1) uptime 0 01:04:12
  RPF route: 10:0:0:1::/64, incoming interface ATM2/3.1001
             neighbor 10:0:0:1::1, owner Local
  Incoming interface list:
      ATM2/3.1001 (10:0:0:1::1/64), Accept/Pim (RPF IIF)
  Outgoing interface list:
      ATM2/0.200 (21:2:2:21::2:1/60), Forward/Pim, 0 01:04:12/never

(10:0:0:1:2::, ff0e::1) uptime 0 01:04:12
  RPF route: 10:0:0:1::/64, incoming interface ATM2/3.1001
             neighbor 10:0:0:1::1, owner Local
  Incoming interface list:
      ATM2/3.1001 (10:0:0:1::1/64), Accept/Pim (RPF IIF)
  Outgoing interface list:
      ATM2/0.200 (21:2:2:21::2:1/60), Forward/Pim, 0 01:04:12/never


Counts: 2 (S, G) entries
        0 (*, G) entries
```

show ipv6 mroute active

- Use to display the active multicast routes with admission bandwidth greater than the specified bandwidth threshold.
- The default is 4000 bps.
- Field descriptions
  - See the **show ipv6 mroute** command and the **show ipv6 mroute summary** command for descriptions of all fields.
- Example 1—Displays active multicast routes with bandwidth above 10000 bps

```
host1#show ipv6 mroute active 10000
Active IP Multicast Routes >=10000 bps

(S, G) uptime d h:m:s[, expires d h:m:s]

[Admission bandwidth: bps]

[QoS bandwidth: bps]

RPF route: addr/mask, incoming interface

neighbor address, owner route-owner

Incoming interface list:

Interface (addr/mask), State/Owner [(RPF IIF)]

Outgoing interface list:

Interface (addr/mask), State/Owner, Uptime/Expires

(52::1, ff3e::1) uptime 0 00:01:07

Admission bandwidth: 47000 bps (adaptive)

QoS bandwidth: 47000 bps (adaptive)

RPF route: 52::/112, incoming interface ATM2/1.17

neighbor 17::2, owner NetmgmtRpf
```

```
Incoming interface list:

ATM2/1.17 (fe80::90:1a00:3140:1ff8/128), Accept/MLD (RPF IIF)

Outgoing interface list:

NULL

Counts: 1 (S, G) entries

0 (*, G) entries
```

- Example 2—Displays the summary of active multicast routes

```
host1#show ipv6 mroute summary active
Active IP Multicast Routes >=4000 bps

Group Address Source Address RPF route RPF Iif #Oifs

--------------- --------------- ------------------ --------------- -----

232.0.0.1 51.0.0.1 51.0.0.0/24 ATM3/1.17 0

232.0.0.2 51.0.0.1 51.0.0.0/24 ATM3/1.17 0

232.0.0.3 51.0.0.1 51.0.0.0/24 ATM3/1.17 0

Counts: 3 (S, G) entries

0 (*, G) entries
```

show mroute port count

- Use to display the mroute port outgoing interface, limits, and counts.

NOTE: This command displays information for mroutes on a port across all virtual routers.

- Field descriptions
  - Port—Slot/port value on the router
  - Limit—None (reserved for future functionality)
  - Count—Number of mroute outgoing interfaces on the specified port
  - BW bps—Bandwidth limit (in bits per second)
  - Admitted—Bandwidth admitted on the port (in bits per second)
- Example

```
host1#show mroute port count

BW     Priority
Port    Limit   Count    bps     BW bps    Hysteresis   Admitted
-----   -----   -----   -----   --------   ----------   --------
1/1/0   None    1       None    None       85           0
1/1/1   None    2       15000   10000      85           2000
```

show ipv6 mroute count

- Use to display information about the number of groups and sources.
- Specify a multicast group address or both a multicast group address and a multicast source address to display information about a particular multicast forwarding entry.
- Field descriptions
  - Counts—Number of types of source group mappings
    - (S,G)—Number of (S,G) entries
    - (*,G)—Number of (*,G) entries
- Example

```
host1#show ipv6 mroute count
IPv6 Multicast Routing Table


Counts: 2000 (S, G) entries
        0 (*, G) entries
```

show ipv6 mroute statistics

- Use to display statistics for packets received through multicast routes that the router has added to the multicast routing table and established on the appropriate line modules.
- Specify a multicast group IPv6 address or both a multicast group IPv6 address and a multicast source IPv6 address to display information about a particular multicast forwarding entry.
- Field descriptions
  - See **show ipv6 mroute** command for descriptions of all fields except the statistics field.
  - Statistics

NOTE: The display shows statistics after the VR has added the multicast route to the multicast routing table and established the route on the appropriate line module. Statistics for interactions before the route is established on the line module are not displayed.

    - Received—Number of packets and bytes that the VR received for this multicast route
    - Forwarded—Number of packets and statistics that the VR has forwarded for this multicast route
    - Rcvd on OIF—Number of packets and statistics that the VR has received on the OIF for this multicast route
- Example

```
host#show ipv6 mroute statistics
IPv6 Multicast Routing Table

(S, G) uptime d h:m:s[, expires d h:m:s]
  [Admission bandwidth: bps]
  [QoS bandwidth: bps]
```

```
  RPF route: addr/mask, incoming interface
             neighbor address, owner route-owner
  Incoming interface list:
      Interface (addr/mask), State/Owner [(RPF IIF)]
  Outgoing interface list:
      Interface (addr/mask), State/Owner, Uptime/Expires

(10:0:0:1:1::, ff0e::1) uptime 0 01:05:23
  Admission bandwidth:
  RPF route: 10:0:0:1::/64, incoming interface ATM2/3.1001
             neighbor 10:0:0:1::1, owner Local
  Incoming interface list:
      ATM2/3.1001 (10:0:0:1::1/64), Accept/Pim (RPF IIF)
  Outgoing interface list:
      ATM2/0.200 (21:2:2:21::2:1/60), Forward/Pim, 0 01:05:23/never
  Statistics:
      Received   : 346 pkts, 22144 bytes
      Forwarded  : 346 pkts, 22144 bytes
      Rcvd on OIF: 0 pkts

(10:0:0:1:2::, ff0e::1) uptime 0 01:05:23
  RPF route: 10:0:0:1::/64, incoming interface ATM2/3.1001
             neighbor 10:0:0:1::1, owner Local
  Incoming interface list:
      ATM2/3.1001 (10:0:0:1::1/64), Accept/Pim (RPF IIF)
  Outgoing interface list:
      ATM2/0.200 (21:2:2:21::2:1/60), Forward/Pim, 0 01:05:26/never
  Statistics:
      Received   : 346 pkts, 22144 bytes
      Forwarded  : 346 pkts, 22144 bytes
      Rcvd on OIF: 0 pkts
```

show ipv6 mroute summary

- Use to display a summary of all or specified multicast routes.
- Specify a multicast group IP address or both a multicast group IP address and a multicast source IP address to display information about a particular multicast forwarding entry.
- Field descriptions
  - Group Address—IP address of the multicast group
  - Source Address—IP address of the multicast source
  - RPF Route—IP address and network mask of the RPF route
  - RPF Iif—Type and identifier for the incoming interface for the RPF route
  - #Oifs—Number of outgoing interfaces
  - Counts—Numbers of types of (S,G) mappings
    - (S,G)—Number of (S,G) entries
    - (*,G)—Number of (*,G) entries

- Example

```
host1#show ipv6 mroute summary
IPv6 Multicast Routing Table

Group Address   Source Address  RPF route          RPF Iif         #Oifs
--------------- --------------- ------------------ --------------- -----
ff0e::1         10:0:0:1:1::    10:0:0:1::/64      ATM2/3.1001     1
ff0e::1         10:0:0:1:2::    10:0:0:1::/64      ATM2/3.1001     1
ff0e::1         10:0:0:1:3::    10:0:0:1::/64      ATM2/3.1001     1
ff0e::1         10:0:0:1:4::    10:0:0:1::/64      ATM2/3.1001     1
ff0e::1         10:0:0:1:5::    10:0:0:1::/64      ATM2/3.1001     1
ff0e::1         10:0:0:1:6::    10:0:0:1::/64      ATM2/3.1001     1
ff0e::1         10:0:0:1:7::    10:0:0:1::/64      ATM2/3.1001     1
ff0e::1         10:0:0:1:8::    10:0:0:1::/64      ATM2/3.1001     1
ff0e::1         10:0:0:1:9::    10:0:0:1::/64      ATM2/3.1001     1
ff0e::1         10:0:0:1:a::    10:0:0:1::/64      ATM2/3.1001     1
ff0e::1         10:0:0:1:b::    10:0:0:1::/64      ATM2/3.1001     1
ff0e::1         10:0:0:1:c::    10:0:0:1::/64      ATM2/3.1001     1
ff0e::1         10:0:0:1:d::    10:0:0:1::/64      ATM2/3.1001     1
ff0e::1         10:0:0:1:e::    10:0:0:1::/64      ATM2/3.1001     1
ff0e::1         10:0:0:1:f::    10:0:0:1::/64      ATM2/3.1001     1

Counts: 16 (S, G) entries
        0 (*, G) entries
```

show ipv6 multicast protocols

- Use to display information about multicast protocols enabled on the router.
- Use the **brief** option to display a summary of information rather than a detailed description.
- Field descriptions
  - Protocol—Name of the multicast protocol
  - Type—Mode of the multicast protocol
    - For PIM—Sparse
    - For MLD—Local
  - Interfaces
    - registered—Number of interfaces on which the protocol is configured
    - owned—Number of interfaces that a protocol owns. If you configure only MLD on an interface, MLD owns the interface. However, if you configure MLD and PIM on the same interface, PIM owns the interface.
  - Registered interfaces—Includes the following information about interfaces on which the protocol is configured
    - Types and identifiers of interfaces. For details about interface types and specifiers, see *JUNOSe Command Reference Guide, About This Guide*.
    - Protocols configured on the interface and the protocol that owns the interface. If you configure only MLD on an interface, MLD owns the interface. However, if you configure MLD and PIM on the same interface, PIM owns the interface.

- Admitted bandwidth / configured admission bandwidth (in bps)
- Number of blocked OIFs
- QoS adjustment bandwidth (in bps)

- Count—Number of multicast protocols on the VR

- Example

```
host1:2#show ipv6 multicast protocols
Multicast protocols:

Protocol Pim
    Type: Sparse
    Interfaces: 1 registered, 1 owned
    Registered interfaces:
        ATM2/1.103 (21:2:2:22::1:2/60) owner Pim
Protocol Mld
    Type: Local
    Interfaces: 1000 registered, 1000 owned
    Registered interfaces:
        ATM2/0.131 (31:2:2:22::2:2/604) local Mld owner Mld
            Admission-bandwidth 2000000/10000000 bps
            QoS Adjust 2000 bps
        ATM2/0.132 (31:2:2:22::2:3/60) local Mld owner Mld
            Admission-bandwidth 0/10000000 bps
            QoS Adjust 0 bps
        ATM2/0.133 (31:2:2:22::2:4/60) local Mld owner Mld
            Admission-bandwidth 8000000/10000000 bps, 2 Blocked OIFs
            QoS Adjust 0 bps
...
Count: 2  protocols
```

show ipv6 multicast protocols brief

- Use to display a summary of information about multicast protocols enabled on the router.
- Field descriptions
  - Protocol—Name of the multicast protocol
  - Registered Interfaces—Number of interfaces on which the protocol is configured.
  - Owned Interfaces—Number of interfaces that a protocol owns. If you configure only MLD on an interface, MLD owns the interface. However, if you configure MLD and PIM on the same interface, PIM owns the interface.
  - Type—Mode of the multicast protocol
    - For PIM—Sparse
    - For MLD—Local
  - Count—Number of multicast protocols on the VR

- Example

```
host1#show ipv6 multicast protocols brief
Protocol  Registered Owned       Type
          Interfaces Interfaces
--------- ---------- ---------- -------------------
Pim                1          1 Sparse
Mld                1          1 Local

Count: 2 protocols
```

show ipv6 multicast routing

- Use to display information about the status of IPv6 multicast on the VR.
- Example

```
host1#show ipv6 multicast routing
Multicast forwarding is enabled on this router
Multicast graceful restart is complete (timer 0 seconds) on this router
Multicast cache-miss processing is enabled on this router
```

BGP Multicast

BGP multicast (MBGP) is an extension of the BGP unicast routing protocol. Many of the functions available for BGP unicasting are also available for MBGP.

The MBGP extensions specify that BGP can exchange information within different types of *address families*. The address families available are unicast IPv4, multicast IPv4, VPN-IPv4, IPv6, and multicast IPv6. When you enable BGP, the router employs unicast IPv4 addresses by default.

You should be thoroughly familiar with BGP before configuring MBGP. See *JUNOSe BGP and MPLS Configuration Guide, Chapter 1, Configuring BGP Routing,* for detailed information about BGP and MBGP.

Chapter 6

Configuring Multicast Listener Discovery

Hosts use Multicast Listener Discovery (MLD) protocol in IPv6 to report their multicast group memberships to neighboring routers. Similarly, multicast routers, such as the E-series router, use MLD to discover which of their hosts belong to multicast groups.

This chapter describes how to configure MLD on an E-series router; it contains the following sections:

Overview

The IPv6 address scheme uses hexadecimal FF at the start of an address for IPv6 multicast. MLD is a protocol that uses these addresses. The following addresses have specific functions:

- You can assign only multicast addresses of global-scope (that is, containing an FF*x*E prefix, where *x* is the flags field) to a multicast group.
- FF02::1 is the link-scope all-nodes address—A packet sent to this address reaches all nodes on a subnetwork.
- FF02::2 is the link-scope all-routers address—A packet sent to this address reaches all routers on a subnetwork.
- FF02::16 is the link-scope all-MLDv2 routers address—A packet sent to this address reaches all MLDv2 routers on a subnetwork.

This implementation of MLD complies with MLD versions 1 and 2. MLDv2 allows for source-specific join and leave messages and is backward compatible with MLDv1. Configuring MLDv1 with the SSM mapping feature provides support for source-specific joins.

MLDv1 mode interfaces exchange the following types of messages between routers and hosts:

- Multicast listener queries
- Multicast listener reports
- Multicast listener done messages

MLDv2 mode interfaces exchange the following types of messages with MLDv2 hosts:

- Multicast listener queries
- MLDv2 multicast listener reports

Multicast Listener Queries

A multicast router can be a querier or a nonquerier. There is only one querier on a network at any time. Multicast routers monitor queries from other multicast routers to determine the status of the querier. If the querier hears a query from a router with a lower IPv6 address, it relinquishes its role to that router.

MLDv1 and MLDv2 mode interfaces send two types of multicast listener queries to hosts on the network:

- General queries to the all-nodes address (FF02::1)
- Specific queries to the appropriate multicast group address

MLDv2 mode interfaces send the following type of queries to MLDv2 hosts:

- General queries
- Group-specific queries
- Source-specific queries

The purpose of a membership group query is to discover the multicast groups to which a host belongs.

MLDv1 and MLDv2 multicast listener queries have a Max Response Time field. This response time is the maximum that a host can take to reply to a query.

Multicast Listener Reports

When a host receives a multicast listener query, it identifies the groups associated with the query and determines to which groups it belongs. The host then sets a timer, with a value less than the Max Response Time field in the query, for each group to which it belongs.

When the timer expires, the host sends a multicast listener report to the group address. When a multicast router receives a report, it adds the group to the membership list for the network and sets a timer to the *multicast address listening interval*. If this timer expires before the router receives another multicast listener report, the router determines that the group has no members left on the network.

If the router does not receive any reports for a specific multicast group within the *maximum response time*, it determines that the group has no members on the network. The router does not forward subsequent multicasts for that group to the network.

MLDv2 supports an extended report format that allows you to report multiple groups and source lists in a single report. These reports are addressed to the all-MLDv2 router's multicast address (FF02::16).

Multicast Listener Done Messages

When an MLDv1 host leaves a group, it sends a multicast listener done message to multicast routers on the network. A host generally addresses multicast listener done messages to the all-routers address, FF02::2.

When an MLDv2 host leaves a group, it sends a multicast listener report. This report includes an empty source list for that group.

Platform Considerations

For information about modules that support MLD on the ERX-7xx models, ERX-14xx models, and the ERX-310 router:

- See *ERX Module Guide, Table 1, Module Combinations* for detailed module specifications.
- See *ERX Module Guide, Appendix A, Module Protocol Support* for information about the modules that support MLD.

For information about modules that support MLD on the E120 router and the E320 router:

- See *E120 and E320 Module Guide, Table 1, Modules and IOAs* for detailed module specifications.
- See *E120 and E320 Module Guide, Appendix A, IOA Protocol Support* for information about the modules that support MLD.

References

For more information about MLD, see the following resources:

- RFC 3710—Multicast Listener Discovery (MLD) for IPv6 (October 1999) on page 566
- IGMP/MLD-based Multicast Forwarding ('IGMP/MLD Proxying')—draft-ietf-magma-igmp-proxy-06.txt (October 2004 expiration) on page 575
- Multicast Group Membership Discovery MIB—draft-ietf-magma-mgmd-mib-06.txt (October 2004 expiration) on page 575

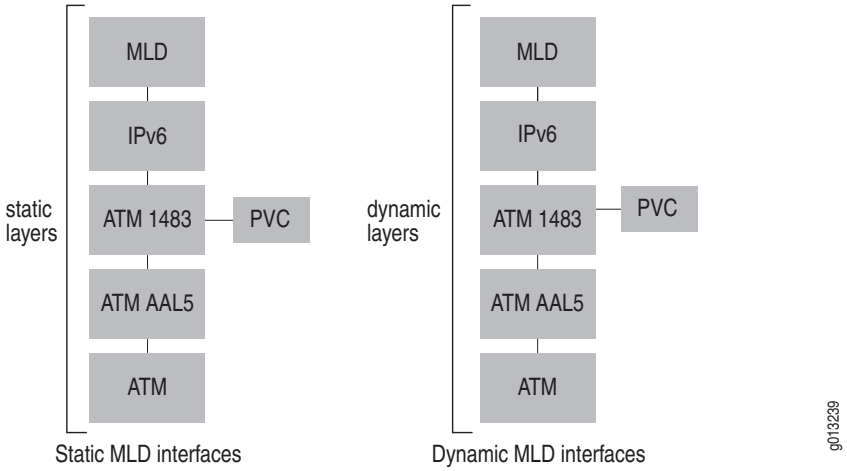
Before You Begin

You can configure MLD only on IPv6 interfaces. For information about configuring IPv6 interfaces, see *JUNOSe IP, IPv6, and IGP Configuration Guide, Chapter 2, Configuring IPv6*.

Configuring Static and Dynamic MLD Interfaces

The router supports *static* and *dynamic* MLD interfaces. Unlike static interfaces, dynamic interfaces are not restored when you reboot the router. For some protocols, dynamic layers can build on static layers in an interface; however, in a dynamic MLD interface, all the layers are dynamic. See Figure 17 for examples of static and dynamic MLD interfaces.

Figure 17: Static and Dynamic MLD Interfaces



Static MLD interfaces are configured with software such as the CLI or an SNMP application; dynamic MLD interfaces are configured with a profile. A profile comprises a set of attributes for an interface; a profile for dynamic MLD interfaces contains attributes for configuring all the layers in the interface.

You define a profile by using the same CLI commands that you use to configure a static MLD interface; however, the mode in which you use the commands differs. Use the commands in Interface Configuration mode to configure a static MLD interface and in Profile Configuration mode to define a profile.

When you have defined a profile, you can apply it to an interface or a group of interfaces. Profiles provide an efficient method of creating and managing large numbers of dynamic interfaces. For detailed information about creating and assigning profiles, see *JUNOSe Link Layer Configuration Guide, Chapter 12, Configuring Dynamic Interfaces*. When you create a profile for dynamic MLD interfaces, specify attributes for configuring all layers in the interface.

You use the MLD commands shown in Table 11 to configure a static MLD interface. You also use these commands to define the attributes for the MLD layer when you create a profile for dynamic MLD interfaces.

Table 11: Static MLD Commands

ipv6 mld	**ipv6 mld query-interval**
ipv6 mld access-group	**ipv6 mld query-max-response-time**
ipv6 mld access-source-group	**ipv6 mld robustness**
ipv6 mld explicit-tracking	**ipv6 mld static-include**
ipv6 mld group limit	**ipv6 mld static-exclude**
ipv6 mld immediate-leave	**ipv6 mld static-group**
ipv6 mld last-member-query-interval	**ipv6 mld version**
ipv6 mld querier-timeout	

The following sections describe the tasks associated with these and other **ipv6 mld** commands.

You can also use various MLD-specific RADIUS attributes in RADIUS Access-Accept messages as an alternative method of configuring certain values. See *JUNOSe Broadband Access Configuration Guide, Chapter 3, Configuring RADIUS Attributes* for additional information.

Enabling MLD on an Interface

You must start MLD on each interface that you want to use the protocol. You can configure MLD and PIM on the same interface. If you configure only MLD on an interface, the router determines that MLD owns that interface. If you configure MLD and PIM on an interface, the router determines that PIM owns the interface.

In an MLDv1 or MLDv2 network, the querier is the router with the lowest IPv6 address.

To start MLD, complete the following steps:

1. Enable MLD on the interface (MLDv2 is the default version).
2. (MLDv1) Specify the MLD version for the interface.

ipv6 mld

- Use to enable MLD on an interface and to set the MLD version to MLDv2. Use the **ipv6 mld version** command to specify a different MLD version.
- Example

  host1:boston(config-if)#**ipv6 mld**

- Use the **no** version to disable MLD on an interface.

ipv6 mld version

- Use to set the MLD version (1 or 2) for the interface.
- Example

  host1:boston(config-if)#**ipv6 mld version 2**

- Use the **no** version to set the version to the default, MLDv2.

Configuring MLD Settings for an Interface

When you start MLD on an interface, it operates with the default settings. You can, however, modify:

- The method that the router uses to remove hosts from multicast groups
- The time interval at which the querier sends multicast listener queries
- The time that a querier waits before sending a new query to hosts from which it receives multicast listener done messages
- The time that a non-querier waits for queries from the current querier before sending query messages to assume responsibility of querier
- The time that a host can take to reply to a query (maximum response time)
- The number of times that the router sends each MLD message from this interface

ipv6 mld immediate-leave

- Use to specify that, when the router receives a multicast listener done message from a host associated with this interface, the router immediately removes that host from the multicast group.

CAUTION: Issue this command only on MLD interfaces to which one MLD host is connected. If more than one MLD host is connected to a LAN through the same interface, and one host sends a done message, the router removes all hosts on the interface from the multicast group. The router loses contact with the hosts that should remain in the multicast group until they send join requests in response to the router's next general multicast listener query.

- Use the MLD-Immediate-Leave RADIUS attribute (VSA 26-100) in RADIUS Access-Accept messages as an alternative method of configuring this value. The RADIUS setting takes precedence over a CLI setting.
- Example

  host1:boston(config-if)#**ipv6 mld immediate-leave**

- Use the **no** version to restore the default behavior, in which the router removes a host from a multicast group if that host does not return a multicast listener report within a certain length of time after receiving a multicast listener query from the router.

ipv6 mld last-member-query-interval

- Use to specify the last-member-query-interval value, in the range 1–255 tenths of a second. When the router receives an MLDv1 leave message or an MLDv2 state change report, it sends out a query and expects a response within the time specified by this value.
- Using a lower value allows members to leave groups more quickly.
- Example

  host1:boston(config-if)#**ipv6 mld last-member-query-interval 90**

- Use the **no** version to restore the default, 10-tenths of a second (1 second).

ipv6 mld querier-timeout

- Use to set the time, in the range 1–400 seconds, that the interface waits for queries from the current querier before sending query messages to assume responsibility of querier.
- Example

  host1:boston(config-if)#**ipv6 mld querier-timeout 200**

- Use the **no** version to set the time to the default, twice the query interval.

ipv6 mld query-interval

- Use to specify how often, in the range 1–300 seconds, the interface sends group membership queries.
- Use the MLD-Query-Interval RADIUS attribute (VSA 26-98) in RADIUS Access-Accept messages as an alternative method of configuring this value. The RADIUS setting takes precedence over a CLI setting.
- Example

  host1:boston(config-if)#**ipv6 mld query-interval 100**

- Use the **no** version to set the polling interval to the default, 125 seconds.

ipv6 mld query-max-response-time

- Use to specify the period in tenths of a second during which the host is expected to respond to a group membership query. The possible period ranges are as follows:
  - IGMPv1 and IGMPv2: 1–255 tenths of a second
  - IGMPv3: 1–31744 tenths of a second
- MLDv1 and MLDv2 include this value in MLD query messages sent out on the interface.
- Using a lower value allows members to join and leave groups more quickly.

- Use the MLD-Query-Max-Resp-Time RADIUS attribute (VSA 26-99) in RADIUS Access-Accept messages as an alternative method of configuring this value. The RADIUS setting takes precedence over a CLI setting.
- Example

  host1:boston(config-if)#**ipv6 mld query-max-response-time 120**

- Use the **no** version to restore the default, 100 tenths of a second (10 seconds).

ipv6 mld robustness

- Use to specify the number of times that the router sends each MLD message from this interface.
- Use a higher value to ensure high reliability from MLD.
- Specify a number in the range 1–4.
- Example

  host1:boston(config-if)#**ipv6 mld robustness 2**

- Use the **no** version to restore the default, 3.

Specifying Multicast Groups

You can use a standard IPv6 access list to specify the multicast groups that a host can join.

ipv6 mld access-group

- Use to restrict hosts on this subnetwork to join only multicast groups that appear on the specified IPv6 access list.
- When configured, the access list is queried whenever the router receives an MLDv1 report requesting membership of a group and MLDv2 ChangeToInclude, IsInclude, ChangeToExclude, or IsExclude reports. The request is ignored if the access list query fails. The **ipv6 mld access-group** command uses IPv6 access lists, which allow both source and destination/group addresses to be specified. You must set the source address to "any."
- Use the MLD-Access-Name RADIUS attribute (VSA 26-74) in RADIUS Access-Accept messages as an alternative method of configuring this value.
- Example

  host1:boston(config-if)#**ipv6 mld access-group boston-list**

- Use the **no** version to dissociate the interface from an access list and to allow hosts on the interface to join any multicast group.

ipv6 mld access-source-group

- Use to restrict hosts on this subnetwork to membership in those (S,G) pairs (also known as "channels") permitted by the specified IPv6 access list.
- When configured, both source and group addresses query the associated access list whenever the router receives an MLDv2 report requesting membership of the (S,G) pairs (that is, the router receives an MLDv2 ChangeToInclude, IsInclude, or AllowNewSource group report). The request is ignored if the access list query fails. The **ipv6 mld access-source-group** command uses IPv6 access lists, which allow both source and destination group addresses to be specified.
- Use the MLD-Access-Src-Name RADIUS attribute (VSA 26-75) in RADIUS Access-Accept messages as an alternative method of configuring this value.
- Example

  host1:boston(config-if)#**ipv6 mld access-source-group dallas-list**

- Use the **no** version to remove any access list restriction.

Assigning a Multicast Group to an Interface

You can assign an interface to send and receive all traffic for a particular multicast group. This feature allows you to control the MLD traffic and to test the behavior of multicast protocols in the network.

ipv6 mld static-group

- Use to send and receive all traffic for a multicast group from a specific interface.
- The interface sets no timers for this group.
- Example

  host1:boston(config-if)#**ipv6 mld static-group ff0e::1**

- Use the **no** version to remove the group from the interface.

Configuring Group Outgoing Interface Mapping

You can configure an MLD protocol interface to use a different outgoing interface (OIF) for multicast-data-forwarding by applying an OIF map. When you configure an OIF map on an MLD protocol interface, the map is applied to all MLD membership requests that the interface receives. To configure OIF mapping on an interface, you first create the OIF map using the **ipv6 mld oif-map** command and then apply that map to an interface with the **ipv6 mld apply-oif-map** command.

To properly configure an interface used in the OIF map for multicast-data-forwarding capability, you must configure the interface version as passive with the **ipv6 mld version** command. You can either specify a passive interface as the OIF or specify the OIF as *self* (to use the MLD protocol interface as the OIF) in the **ipv6 mld oif-map** command.

ipv6 mld apply-oif-map

- Use to apply the specified outgoing interface (OIF) map to the current interface.
- Example

  host1(config-subif)#**ipv6 mld apply-oif-map OIFMAP**

- Use the **no** version to remove the outgoing interface map association from the interface.

ipv6 mld oif-map

- Use to create an OIF map.
- Use the MLD-OIF-Map-Name RADIUS attribute (VSA 26-76) in RADIUS Access-Accept messages as an alternative method of configuring this value.
- Example

host1(config)#**ipv6 mld oif-map OIFMAP atm 3/0.1 ff0e::1:1/128 2001::1:1/128**
host1(config)#**ipv6 mld oif-map OIFMAP atm 3/0.2 ff0e::1:1/128 2001::1:1/128**
host1(config)#**ipv6 mld oif-map OIFMAP atm 3/0.3 ff0e::1:1/128**
host1(config)#**ipv6 mld oif-map OIFMAP atm 3/0.4 ff0e::1:0/112 2001::1:1/128**
host1(config)#**ipv6 mld oif-map OIFMAP atm 3/0.5 ff0e::1:0/112 2001::1:1/128**
host1(config)#**ipv6 mld oif-map OIFMAP self ::/0 2001::1:0/112**

- Use the **no** version to remove an outgoing interface map attribute.

ipv6 mld version

- Use to set the MLD version (1 or 2) for the interface or specify a passive interface with only multicast-data-forwarding capability (passive).
- Use the MLD-Version RADIUS attribute (VSA 26-77) in RADIUS Access-Accept messages as an alternative method of configuring this value.
- Example

  host1:dallas(config-if)#**ipv6 mld version passive**

- Use the **no** version to set the version to the default, mldv2.

Configuring SSM Mapping

SSM mapping enables the router to determine one or more source addresses for group G. The mapping effectively translates an MLDv1 multicast listener report to an MLDv2 report, enabling the router to continue as if it had initially received an MLDv2 report. After the router is joined to these groups, it sends out PIM join messages and continues to enable joining from these groups, as long as it continues to receive MLDv1 membership reports and no change occurs to the SSM mapping for the group.

When you statically configure SSM mapping, the router can discover source addresses from a statically configured table.

The following applies when you configure SSM mapping:

- When enabled, and either you have not configured a static SSM map or the router cannot find any matching access lists, the router continues to accept (*,G) groups. The PIM SSM range must deny any unacceptable SSM group addresses.
- When you issue the **no ipv6 mld ssm-map enable** command, the router removes all SSM map (S,G) states and establishes a (*,G) state.
- You can enter multiple **ssm-map static** commands for different access lists. Also, you can enter multiple **ssm-map static** commands for the same access list, as long as the access list uses different source addresses.
- SSM maps do not process statically configured groups.

ipv6 mld ssm-map enable

- Use to enable SSM mapping on the router. SSM mapping statically assigns sources to MLDv1 groups. You must use SSM mapping for MLDv1 hosts to interoperate with PIM SSM. SSM mapping allows the router use a statically configured list to translate (*,G) memberships to (S,G) memberships.
- Example

  host1:boston(config)#**ipv6 mld ssm-map enable**

- Use the **no** version to disable the SSM map.

ipv6 mld ssm-map static

- Use to specify an access list and source address for use in SSM mapping. SSM mapping statically assigns sources to MLDv1 groups. You must use SSM mapping for MLDv1 hosts to interoperate with PIM SSM. SSM mapping allows the router to use a statically configured list to translate (*,G) memberships to (S,G) memberships.
- The **ipv6 mld ssm-map static** command uses IPv6 access lists, which allow both source and destination/group addresses to be specified. You must set the source address to "any."
- Example

  host1:boston(config)#**ipv6 mld ssm-map static boston-list 2001::1**

- Use the **no** version to remove the SSM map association.

Limiting the Number of Accepted MLD Groups

By default, there is no limit on the number of MLD groups that an MLD interface can accept. However, you can manage multicast traffic on the router by restricting the number of MLD groups accepted by:

- A specific port on an I/O module
- A specific MLD interface

If you set limits for both a port and interfaces on that port, the router uses the lower of the two limits when determining how many MLD groups an interface can accept. For example, if you set a limit of 10 groups for the port and 15 groups for each interface, the router allows only 10 groups to be accepted among the interfaces.

However, if you set a limit for a port and that limit is lower than the number of groups currently accepted by the interfaces on that port, the router does not dissociate the groups from the interfaces. The router enforces the new limit on the port when the number of groups associated with the interfaces falls to that limit. For example, if the interfaces on the port have accepted a total of 15 groups, and you set a limit of 10 groups on the port, the router does not disconnect any of the groups and does not allow the interfaces to accept any more groups. Over time, some groups leave the interfaces and, eventually, a maximum of ten groups remains connected.

ipv6 mld group limit

- Use to limit the number of MLD groups that an interface can accept.
- Example

  host1:boston(config-if)#**ipv6 mld group limit 5**

- Use the **no** version to restore the default situation, in which there is no limit on the number of MLD groups that an interface can accept.

multicast group port limit

- Use to limit the number of MLD groups that a port can accept.
- Specify the identifier for the port in *slot/port* format (ERX routers) or in *slot/adapter/port* format (E120 and E320 routers) and the maximum number of MLD groups that interfaces can accept.
  - *slot*—Number of the chassis slot in the range 0–6 (ERX-7xx models), 0–13 (ERX-14xx models), 0–5 (E120 router), or 0–16 (E320 router)
  - *adapter*—Number of the bay in which the I/O adapter (IOA) resides. This identifier applies to the E120 and E320 routers only. In the software, adapter 0 identifies the right IOA bay (E120 router) and the upper IOA bay (E320 router); adapter 1 identifies the left IOA bay (E120 router) and the lower IOA bay (E320 router).
  - *port*—Port number on the I/O module or IOA
- Example 1—ERX models

  host1(config)#**multicast group port 3/0 limit 5**

- Example 2—E120 and E320 routers

  host1(config)#**multicast group port 3/1/0 limit 5**

- Use the **no** version to restore the default situation, in which there is no limit on the number of MLD groups that a port can accept.

Including and Excluding Traffic

MLDv2 extends MLDv1 functionality with the ability to include or exclude specific multicast traffic sources. That is, with MLDv2, hosts signal (S,G) pairs that they want to include or exclude.

For hosts that cannot signal group membership dynamically, you can use the **ipv6 mld static-include** or **ipv6 mld static-exclude** command to statically include or exclude multicast traffic, respectively.

MLDv2 is the industry-designated standard protocol for hosts to signal channel subscriptions in SSM. For additional information about SSM, see *PIM Source-Specific Multicast* on page 83.

ipv6 mld static-exclude

- Use to statically exclude the MLD (S,G) membership for a host that is not capable of dynamically signaling group membership.
- Example

  host1:boston(config-if)#**ipv6 mld static-exclude 2001::1 ff0e::1**

- Use the **no** version to remove the static designation.

ipv6 mld static-include

- Use to statically include the MLD (S,G) membership for a host that is not capable of dynamically signaling group membership.
- Example

  host1:boston(config-if)#**ipv6 mld static-include 2001::1 ff0e::1**

- Use the **no** version to remove the static designation.

Configuring Explicit Host Tracking

Explicit host tracking enables the router to explicitly track each individual host that is joined to a group or channel on a particular multi-access network.

Explicit host tracking provides the following benefits:

- Minimal leave latency when a host leaves a multicast group or channel. When the router receives a leave message for a group or channel on an interface, it accesses a list of hosts and immediately stops forwarding traffic if the sender is the last host to request traffic for that group or channel. The leave latency is bound only by the packet transmission latencies in the multi-access network and the processing time in the router.
- Ability to change channels quickly in networks where bandwidth is constrained between a multicast-enabled router and hosts.
- Ability to determine what multicast hosts are joined to particular multicast groups or channels; this is useful for accounting purposes.
- Reduction of control message traffic on the network because, when it receives a leave message, the router no longer needs to send out MLD queries to verify membership. As a result, interested hosts also do not need to respond to these queries with reports.
- Tracking based on MLD reports for hosts in both include and exclude modes for every multicast group or channel on an interface.

When the router is configured for explicit host tracking and starts performing immediate leave using the host information collected, every leave message received for a group or channel is treated as follows:

- The router checks the number of hosts that receive traffic from this group or channel.
- If the host sending the leave message is the only host, it performs immediate leave for that group or channel on that interface. The router removes the interface from the multicast group or channel immediately, without sending out a group or group-source specific query and waiting for the last member query interval.
- If the host sending the leave message is not the only host receiving traffic for that group or channel, the router removes the host from the list of hosts on that interface, but keeps the interface in the outgoing interface list for the multicast group or channel. No group or group-source specific queries are sent.

You can enable MLD explicit host tracking on an interface only if MLD V1 or V2 has been previously enabled on the interface. Explicit host tracking is not enabled by default when you enable MLD on the interface. Explicit host tracking cannot be configured on passive MLD interfaces.

When you enable explicit host tracking on an interface that has a membership state, the router does not immediately start performing immediate leave. For a maximum of group membership interval seconds, the router only performs host tracking. Any leave messages that the router receives during this period receive normal leave processing. Any leave messages received after this interval has elapsed receive immediate leave processing, when appropriate.

When explicit host tracking is enabled on an MLD V2 interface, even if a group has to downgrade to MLD V1 due to the presence of an MLD V1 host, explicit host tracking continues for that group. To avoid this, you can use the **disable-if-mld-v1-detected** keyword. If you select this option, the router turns off explicit host tracking for the group when MLD V1 host reports are received for the group on that interface. This option does not have any significance on an interface configured for MLD V1 and is ignored if selected.

If you execute the command on an interface that was previously enabled for immediate-leave, the configuration is accepted, immediate-leave is turned off and an appropriate warning message logged. Any attempt to configure immediate-leave on an interface that has explicit host tracking enabled is rejected and an error message logged.

The following example enables MLD V2 explicit host tracking on interface 3/0.101 with the default configuration where the router continues to perform explicit host tracking for MLD V1 groups. To override this default configuration, use the **ip mld explicit-tracking disable-if-mld-v1-detected** command.

```
interface 3/0.101
ip mld version 2
ip mld explicit-tracking
end
```

ipv6 mld explicit-tracking

- Use to set explicit host tracking for MLD interfaces.
- To disable explicit host tracking if MLD V1 hosts are detected, use the **disable-if-mld-detected** keyword.
- Example

  host1(config)#**ipv6 mld explicit-tracking**

- Use the **no** version to disable explicit host tracking on the interface. Use the **no** version with the **disable-if-mld-detected** keyword to revert to the default explicit host tracking behavior.

Disabling and Removing MLD

You can disable and reenable MLD on the VR. You can also remove MLD from the VR and re-create it on the VR.

mld disable

- Use to disable MLD on a VR.
- Example

  host1(config)#**virtual-router boston**
  host1:boston(config)#**router mld**
  host1:boston(config-router)#**mld disable**

- Use the **no** version to enable MLD on a VR.

router mld
ipv6 router mld

- Use to create and enable MLD on a VR or to access MLD Router Configuration mode.
- Example 1

  host1(config)#**virtual-router boston**
  host1:boston(config)#**router mld**

- Example 2

  host1(config)#**virtual-router boston**
  host1:boston(config)#**ipv6 router mld**

- Use the **no** version to delete MLD and MLD proxy from the VR.

Monitoring MLD

You can establish a reference point for MLD statistics by setting the statistics counters to zero.

To display MLD parameters, use the **show** commands described in this section.

NOTE: The E120 router and E320 router output for **monitor** and **show** commands is identical to output from other E-series routers, except that the E120 and E320 router output also includes information about the adapter identifier in the interface specifier (*slot/adapter/port*).

baseline ipv6 mld

- Use to set the counters for MLD statistics to zero.
- Example

  (host1)#**baseline ipv6 mld**

- There is no **no** version.

show ipv6 mld

- Use to display MLD information for a VR.
- Field descriptions
  - Administrative state—Status of MLD in the software: enabled or disabled
  - Operational state—Status of MLD on the VR: enabled or disabled
  - total interfaces—Number of interfaces on which you started MLD
  - enabled—Number of interfaces on which MLD is enabled
  - disabled—Number of interfaces on which MLD is disabled
  - learned groups—Number of multicast groups that the VR has discovered
  - MLD Statistics Rcvd—Statistics for MLD messages received
    - total—Number of MLD messages received
    - checksum errors—Number of MLD messages received with checksum errors
    - unknown types—Number of messages received that are not multicast listener queries, multicast listener reports, or multicast listener done messages
    - discards—Number of multicast listener discards
    - queries—Number of multicast listener queries
    - reports—Number of multicast listener reports
    - leaves—Number of done messages
  - MLD Statistics Sent—Number of multicast listener queries sent
- Example

```
host1:boston#show ipv6 mld
Routing Process MLD, Administrative state enabled, Operational state enabled
    4 total interfaces, 4 enabled, 0 disabled
    2 learned groups
MLD Statistics:
    Rcvd: 3 total, 0 checksum errors, 0 unknown types, 0 discards
          0 queries, 3 reports, 0 leaves
    Sent: 5 total
```

show ipv6 mld groups

- Use to display statically joined and directly connected groups learned through MLD.
- Field descriptions
  - Grp Address—Address of the multicast group
  - Interface—Interface that discovered the multicast group
  - oif-map—Name of the OIF map and the mapped OIF interface, if a group or source has been mapped to an OIF
  - State—MLD version of the group
  - Reporter—Link-local address of the host reporting the multicast group
  - ExpTim—Remaining time, in seconds, at which the router stops polling for more members of this group
  - oldHTo—Remaining time at which the router stops polling for more MLDv1 members of a group. If this value is 0, the interface has received no MLDv1 reports for the group.
  - Included Sources—Sources included in the multicast group
  - Excluded Sources—Sources excluded from the multicast group
  - Counts—Number of source-group mappings by version and state
- Example 1—Without OIF mapping

```
host1:boston#show ipv6 mld groups
    Grp Address           Interface          State      Reporter     ExpTim oldHTo
------------------- --------------------- -------- --------------- ------ ------
ff0e::1             ATM2/0.15             Version2 fe80::90:1a02:1 54     0
                                                   640:91d
ff0e::4:1           ATM2/0.15             Version2 fe80::90:1a02:1 54     0
                                                   640:91d
  Included Sources:
    51::1                                                          54
    51::2                                                          54

Counts: 2 version-2, 0 version-1, 0 check state, 0 disabled
       (2 total)
        0 excluded
        Source-groups: 2 included, 0 excluded
```

- Example 2—With OIF mapping

```
host1:boston#show ipv6 mld groups
    Grp Address           Interface          State      Reporter     ExpTim oldHTo
------------------- --------------------- -------- --------------- ------ ------
ff3e::1             ATM5/0.12             Version2 fe80::f7:0:91a: 377    0
                                                   0
                    oif-map OIFMAP ATM5/0.
                    121
ff3e::1             ATM5/0.13             Version2 fe80::f7:0:a1a: 369    0
                                                   0
                    oif-map OIFMAP ATM5/0.
                    121
ff3e::2             ATM5/0.12             Version2 fe80::f7:0:91a: 370    0
                                                   0
```

```
  Included Sources:
    10::2           oif-map OIFMAP self                          370
    10::10          oif-map OIFMAP ATM5/0.                       370
                    120
    10::11          oif-map OIFMAP ATM5/0.                       370
                    121
ff3e::2             ATM5/0.13             Version2 fe80::f7:0:a1a: 373    0
                                                   0
  Included Sources:
    10::2           oif-map OIFMAP self                          373
    10::10          oif-map OIFMAP ATM5/0.                       373
                    120
    10::11          oif-map OIFMAP ATM5/0.                       373
                    121

Counts: 4 version-2, 0 version-1, 0 check state, 0 disabled
       (4 total)
        0 excluded
        Source-groups: 6 included, 0 excluded
```

show ipv6 mld interface

- Use to display MLD information for interfaces on which you enabled MLD.
- Specify the **brief** keyword to see a summary of the information.
- Specify the **count** keyword to see the number of MLD interfaces.
- Field descriptions
  - Interface—Type of interface and interface specifier. For details about interface types and specifiers, see *JUNOSe Command Reference Guide, About This Guide*.
  - address—IPv6 link-local address of the interface
  - Administrative state—Status of the interface in the software: enabled or disabled
  - Operational state—Physical status of the interface: enabled or disabled
  - Version—MLD version
  - State—Function of the interface: querier or nonquerier
  - Query Interval—Time interval at which this interface sends query messages
  - Other querier present interval—Time that the interface waits before declaring itself as the querier
  - Maximum response time—Time interval during which this interface expects a host to respond
  - Last member query interval—Time that this interface waits before sending a new query to a host that sends a group leave message
  - Robustness—Number of times this interface sends MLD messages
  - Information about IPv6 access lists configured with the **ipv6 mld access-group** command
    - Inbound access group—Access list specified
    - No inbound access group—No access list specified

- Information about IPv6 access lists configured with the **ipv6 mld access-source-group** command
  - Inbound access source-group—Access list specified
  - No inbound access source-group—No access list specified
- Information about OIF maps configured with the **ipv6 mld apply-oif-map** command
  - Inbound apply-oif-map—Map name specified
  - No inbound apply-oif-map—No map name specified
- Immediate Leave—Setting of the **ipv6 mld immediate-leave** command: enabled or disabled
- Explicit Host Tracking—Setting of the **ipv6 mld explicit-tracking** command: enabled or disabled
- Max-Group limit—Number of MLD groups that the interface can accept, as configured with the **ipv6 mld group limit** command
- Group Count—Number of MLD groups that the interface has accepted
- IOA packet replication—Hardware multicast packet replication interface to which egress multicast packets on this interface are redirected
- Interface statistics Rcvd—Information about MLD messages received on this interface
  - reports—Number of group multicast listener reports received
  - leaves—Number of group multicast listener done messages received
  - wrong version queries—Number of multicast listener queries received from devices running a different version of MLD
- Interface statistics Sent—Number of MLD messages this interface has sent
- Interface statistics Groups learned—Number of groups this interface has discovered
- Counts—Total number of MLD interfaces

- Example

```
host1:boston#show ipv6 mld interface

Interface ATM5/0.1 address fe80::f7:0:321a:0
Administrative state enabled, Operational state enabled
Interface parameters:
    Version 1
    State Querier
    Query Interval 125 secs, 123 secs before the next query
    Other querier present interval 250 secs
    Maximum response time 100 (in 10ths of a second)
    Last member query interval 10 (in 10ths of a second)
    Robustness 3
    No inbound access group
    No inbound access source-group
    No inbound apply-oif-map
    Immediate Leave: disabled
    Explicit Host Tracking: enabled
    Max-Group limit: No Limit
    Group Count: 0
```

```
Interface statistics:
    Rcvd: 0 reports, 0 leaves, 0 wrong version queries
    Sent: 14 queries
    Groups learned: 0

Counts: 0 down, 0 init state, 1 querier, 0 non-querier, 1 Total
```

- Example 2

```
host1#show ipv6 mld interface gigabitEthernet 3/0.0
Interface GigabitEthernet3/0.0 address 10.1.1.1/255.255.255.0
Administrative state enabled, Operational state enabled
Interface parameters:
    Version 1
    State Querier
    Query Interval 125 secs, 123 secs before the next query
    Other querier present interval 250 secs
    Maximum response time 100 (in 10ths of a second)
    Last member query interval 10 (in 10ths of a second)
    Robustness 3
    No inbound access group
    No inbound access source-group
    No inbound apply-oif-map
    Immediate Leave: disabled
    Explicit Host Tracking: enabled
    Max-Group limit: No Limit
    Group Count: 0
    IOA packet replication gigabitEthernet 3/8.1
Interface statistics:
    Rcvd: 0 reports, 0 leaves, 0 wrong version queries
    Sent: 14 queries
    Groups learned: 0

Counts: 0 down, 0 init state, 1 querier, 0 non-querier, 1 Total
```

show ipv6 mld interface brief

- Use to display a summary of MLD information for interfaces on which you enabled MLD.
- Field descriptions
  - Interface—Type of interface and interface specifier. For details about interface types and specifiers, see *JUNOSe Command Reference Guide, About This Guide*.
  - Intf Address—IPv6 link-local address of the interface
  - Ver—MLD version
  - State—Function of the interface: querier or nonquerier
  - Querier—IPv6 address of the querier on the network to which this interface connects
  - QTime—Remaining time interval at which this interface sends query messages
  - QPTime—Remaining time that the interface waits before declaring itself as the querier

- Example

```
host1:boston#show ipv6 mld interface brief

   Interface     Intf Address   Ver    State        Querier     QTime  QPTime
--------------- --------------- --- ----------- --------------- ------ ------
ATM5/0.1        fe80::f7:0:231a 1   Querier     fe80::f7:0:231a 1      0
                :0                              :0
ATM5/0.200      fe80::f7:0:231a 2   Querier     fe80::f7:0:231a 20     0
                :0                              :0

Counts: 0 down, 0 init state, 2 querier, 0 non-querier, 2 Total
```

show ipv6 mld mapped-oif

- Use to display the current mappings to all mapped outgoing interfaces or to the specified mapped outgoing interface.
- Field descriptions
  - OIF—Outgoing interface used in an OIF map
  - Oper—Operation status of the outgoing interface
  - Group Address—Multicast group IP address associated with the OIF
  - Source Address—Source IP address associated with the OIF
  - Join I/F—MLD protocol interface associated with the OIF
  - Map Name—Name of the map associated to the OIF
  - Counts—Number of source-group mappings to OIFs
- Example

```
host1#show ipv6 mld mapped-oif
      OIF       Oper  Group Address  Source Address     Join I/F      Map Name
--------------- ---- --------------- --------------- --------------- ----------
ATM5/0.120      Up   ff3e::2         10::10          ATM5/0.12       OIFMAP
                                                     ATM5/0.13       OIFMAP
ATM5/0.121      Up   ff3e::1         *               ATM5/0.12       OIFMAP
                                                     ATM5/0.13       OIFMAP
                     ff3e::2         10::11          ATM5/0.12       OIFMAP
                                                     ATM5/0.13       OIFMAP

Counts: 3 source-group mappings
```

show ipv6 mld oif-map

- Use to display all outgoing interface (OIF) maps or the OIF map for the specified map name.
- Field descriptions
  - Map Name—Name of the map associated to the show output
  - Group Prefix—Multicast group IPv6 prefix
  - Source Prefix—Source IPv6 prefix
  - OIF—Outgoing interface associated with the group and source prefix
- Example

```
host1#show ipv6 mld oif-map
     Map Name         Group Prefix      Source Prefix           OIF
------------------ ------------------ ------------------ ------------------
OIFMAP             ff3e::/112         ::/0               ATM5/0.121
                   ff3e::/112         10::2/128          self
                   ff3e::/112         10::10/128         ATM5/0.120
                   ff3e::3/128        ::/0               ATM5/0.130
                   ff3e::4/128        ::/0               ATM5/0.130
```

show ipv6 mld membership

- Use to display MLD membership information for multicast groups and (S, G) channels.
- Specify the **brief** keyword to see a summary of the information.
- Specify the **tracked** keyword to see interface information only for interfaces where explicit host tracking is enabled.
- Field descriptions
  - Group—Multicast group or (S, G) channel
  - Source—(S, G) entries that are forwarding traffic
  - Reporter—Hosts that requested including sources or that have not requested excluding sources. If listed under a group, host that sent exclude reports for the group. If listed under a source, host that requested traffic from this source for the group. For any (S, G), if listed under a source, indicates hosts interested in the traffic for this (S, G).

- ExpTim—Expiration time
- Flags
  - M—Uses Oifmap
  - S—SSM mapped
  - T—Tracked
  - 1, 2—MLD version that the group is in
- Interface—Type of interface and interface specifier. For details about interface types and specifiers, see *JUNOSe Command Reference Guide, About This Guide*.

- Example

```
host1:boston#show ipv6 mld membership
Flags: M - Uses Oifmap  S- SSM mapped  T - tracked
       1,2 - The version of MLD the group is in
Reporter:
    <ip-address> - last reporter if the group is not explicitly tracked
    <n>/<m> - <n> reporters include mode, <m> reporters in exclude

Group      Source      Reporter                 ExpTim    Flags     Interface
---------  --------    ----------------------   ------    ------    ---------------
ff0e::40      *        fe80::90:1a02:1640:91d   02:41     2S        FastEthernet2/1

ff0e::50               1/2                      02:56    3MT       FastEthernet2/2
                       fe80::90:1a02:1640:911   02:30
                       fe80::90:1a02:1640:912   02:48
           20::11
                       fe80::90:1a02:1640:913   02:56
           20::12
                       fe80::90:1a02:1640:911   02:30
           20::13
                       fe80::90:1a02:1640:911   02:30
                       fe80::90:1a02:1640:912   02:48
                       fe80::90:1a02:1640:913   02:56
ff0e::60               fe80::90:1a02:1640:901   01:56     3         FastEthernet2/3
           10::10                               02:45
           10::11                               02:35
           10::12                               02:15
           10::14                               stop
ff0e::70               fe80::90:1a02:1640:91    stop      3         FastEthernet2/4
           40::10                               01:10
           40::11                               01.24
ff0e::80               2/0                      stop      3T        FastEthernet2/5
           50::10
                       fe80::90:1a02:1650:910   02:48
           50::11
                       fe80::90:1a02:1650:920   02:56
                       fe80::90:1a02:1650:910   02:48
           50::12      fe80::90:1a02:1650:920   02:56
ff0e::90               0/3                      02:56     2T        FastEthernet2/6

              *
                       fe80::90:1a02:1660:910   02:48
                       fe80::90:1a02:1660:920   02:56
                       fe80::90:1a02:1660:930   02:48
```

show ipv6 mld oif-mapping

- Use to display the mapped OIF to be assigned to a given map-name, group address, and source address.
- Field descriptions
  - OIF-MAP Name—Name of the map requested
  - Group Address—Multicast group IP address requested
  - Source Address—Source IP address requested
  - Mapped OIF—Join interface associated with the OIF map
- Example

```
host1#show ipv6 mld oif-mapping OIFMAP ff3e::1 10::10
    OIF Mapping
    OIF-MAP Name   : OIFMAP
    Group Address  : ff3e::1
    Source Address : 10::10
    Mapped OIF     : ATM5/0.120
```

show ipv6 mld ssm-mapping

- Use to display the SSM mapping state and the source list mapping associated with a multicast group address.
- Field descriptions
  - SSM Mapping—Status of SSM mapping on the interface (enabled or disabled)
  - Group Address—Multicast group address requested
  - Source List—List of sources mapped to the multicast group address
- Example

```
host1:boston#show ipv6 mld ssm-mapping ff3e::1
    SSM Mapping   : Enabled
    Group Address : ff3e::1
    Source List   : 2001::1
                  : 2001::2
```

show multicast group limit

- Use to display the number of MLD groups that ports have accepted and, if configured, the maximum number of groups that ports can accept.
- A value of –1 indicates that no port group limit is configured.
- Only ports that have accepted MLD groups and ports for which you have configured a limit for the number of MLD groups appear in this display.

- Field descriptions
  - Port—Identifier of the port in *slot*/*port* format
    - *slot*—Number of the chassis slot in the range 0–6 (ERX-7xx models) or 0–13 (ERX-14xx models)
    - *port*—Port number on the I/O module
  - limit—Maximum number of MLD groups that the port can accept. A value of –1 indicates that no limit has been specified.
  - count—Actual number of MLD groups the port has accepted
- Example

```
host1:boston#show multicast group limit
Port      limit count
--------- ----- -----
2/0           5     0
2/1          -1     1
```

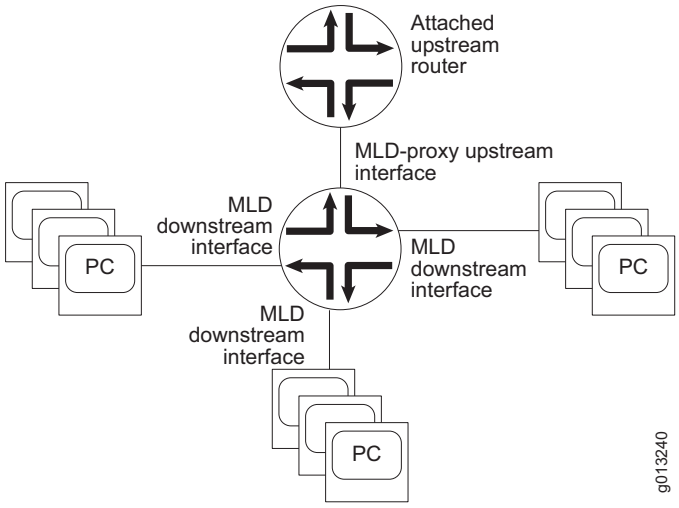
MLD Proxy Overview

MLD proxy enables the router to issue MLD host messages on behalf of hosts that the router discovered through standard MLD interfaces. The router acts as a *proxy* for its hosts. The E-series router supports MLD proxy versions 1 and 2.

Figure 18 shows a router in an MLD proxy configuration. You enable MLD proxy on one interface, which connects to a router closer to the root of the tree. This interface is the *upstream interface*. The attached upstream router on the upstream interface should be running MLD.

You enable MLD on the interfaces that connect the router to its hosts that are farther away from the root of the tree. These interfaces are known as *downstream interfaces*.

Figure 18: Upstream and Downstream Interfaces



As described in *Overview* on page 174, hosts interact with the router through the exchange of MLD messages. Similarly, when you configure MLD proxy, the router interacts with the router on its upstream interface through the exchange of MLD messages. However, when acting as the proxy, the router performs the host portion of the MLD task on the upstream interface as follows:

- When queried, sends multicast listener reports to the group.
- When one of its hosts joins a multicast address group to which none of its other hosts belong, sends unsolicited multicast listener reports to that group.
- When the last of its hosts in a particular multicast group leaves, the group sends either an unsolicited multicast listener done report to the all-routers address (FF02::2) for MLDv1 or an MLDv2 multicast listener report to the all-MLDv2 routers address (FF02::16).

Configuring MLD Proxy

To configure a downstream interface, enable MLD on that interface. To configure MLD proxy on the router, complete the following tasks:

1. Enable IPv6 multicast.
2. Identify the interface that you want to act as the upstream interface.
3. Enable MLD proxy on that interface.
4. (Optional) Specify how often the router should send unsolicited reports to routers on the upstream interface.

ipv6 mld-proxy

- Use to enable MLD proxy on an interface.
- The interface for which you enable MLD proxy is the upstream interface.

NOTE: You can enable only one upstream interface.

- You can specify either MLD proxy version 1 or 2. The default is version 2.
- Example

  host1(config-if)#**ipv6 mld-proxy**

- Use the **no** version to disable MLD proxy on an interface.

ipv6 mld-proxy unsolicited-report-interval

- Use to specify, in tenths of a second, how often the upstream interface should transmit unsolicited reports.

NOTE: Issue this command only on the upstream interface. Otherwise, this command has no effect.

- Example

  host1(config-if)#**ipv6 mld-proxy unsolicited-report-interval 600**

- Use the **no** version to transmit unsolicited reports using the default value, 100-tenths of a second (10 seconds).

ipv6 mld-proxy version

- Use to set the MLD proxy version for the interface.
- Example

  host1(config-if)#**ipv6 mld-proxy version 1**

- Use the **no** version to set the version to its default value, MLDv2.

Setting the MLD Proxy Baseline

You can set the counters for the numbers of queries received and reports sent on the upstream interface to zero. This feature allows you to establish a reference point for MLD proxy statistics.

baseline ipv6 mld-proxy interface

- Use to set the counters for the numbers of queries received and reports sent on the upstream interface to zero.

NOTE: Issue this command only on the upstream interface. Otherwise, this command will have no effect.

- Example

  (host1)#**baseline ipv6 mld-proxy interface**

- There is no **no** version.

Monitoring MLD Proxy

To display MLD proxy parameters, use the following **show** commands.

show ipv6 mld-proxy

- Use to display MLD proxy parameters for a VR.
- Field descriptions
  - Routing Process—MLD proxy protocol
  - Administrative state—State of MLD proxy in the software
  - Operational state—Operational state of MLD proxy: enabled or disabled
  - total interfaces—Number of MLD proxy interfaces on the VR; currently only one upstream interface per VR
  - state—Operational state of the MLD proxy interfaces: enabled or disabled
  - multicast group—Number of multicast groups associated with MLD proxy interfaces
- Example

```
host1#show ipv6 mld-proxy
Routing Process MLD Proxy, Administrative state enabled, Operational state
enabled
    total 1 upstream interface, state enabled
    1 multicast group
```

show ipv6 mld-proxy groups

- Use to display information about multicast groups that MLD proxy reported.
- Field descriptions
  - Grp Address—Address of the multicast group
  - Interface—Type and identifier of the upstream interface associated with the multicast group
  - Grp Mode
    - Blank—No sources included or excluded for this group
    - Include—Sources included for this group
    - Exclude—Sources excluded for this group
  - Count—Total number of multicast groups associated with this interface
- Example 1

```
host1#show ipv6 mld-proxy groups

Grp Address     Interface       Grp Mode
--------------- --------------- --------------
ff0e::1         ATM5/1.200
ff0e::2         ATM5/1.200
ff0e::3         ATM5/1.200      Include(1):
  2001::1
```

```
ff0e::4         ATM5/1.200
ff0e::5         ATM5/1.200      Exclude(1):
  2001::2

Counts: 3 <*,G>, 1 Exclude (1 sources), 1 Include (1 sources)
       (5 total)
```

- Example 2

```
host1#show ipv6 mld-proxy groups ff0e::1
Grp Address     Interface       Grp Mode
--------------- --------------- --------------
ff0e::1         ATM5/1.200

Counts: 1 <*,G>
       (1 total)
```

- Example 3

```
host1#show ipv6 mld-proxy groups count
Counts: 3 <*,G>, 1 Exclude (1 sources), 1 Include (1 sources)
       (5 total)
```

show ipv6 mld-proxy interface

- Use to display information about the interface on which you configured MLD proxy.
- To view information about a particular interface, enter an interface type and specifier, such as atm 3/0. For details about interface types and specifiers, see *JUNOSe Command Reference Guide, About This Guide*.
- Specify the **brief** option to display a summary rather than a detailed description.
- Field descriptions
  - Interface—Type of upstream interface. For details about interface types, see *JUNOSe Command Reference Guide, About This Guide*.
  - Address—Address of upstream interface
  - Administrative state—State of upstream interface in the software: enabled or disabled
  - Operational state—Physical state of upstream interface: enabled or disabled
  - Version—MLD version on this interface
  - State—Presence of MLDv1 routers on the same subnet as this upstream interface
  - Unsolicited report interval—Time interval at which this upstream interface sends unsolicited group membership report
  - multicast group—Number of multicast groups associated with this upstream interface

  - Interface statistics Rcvd—Statistics for messages received on this interface
    - v1 queries—Number of MLDv1 multicast listener queries received
    - v1 report—Number of MLDv1 multicast listener reports received
    - v2 queries—Number of MLDv2 multicast listener queries received
    - v2 report—Number of MLDv2 multicast listener reports received
  - Interface statistics Sent—Statistics for messages sent from this interface
    - v1 reports—Number of MLDv1 multicast listener reports sent
    - v1 leaves—Number of multicast listener done messages sent
    - v2 reports—Number of MLDv2 multicast listener reports sent

- Example 1

```
host1#show ipv6 mld-proxy interface

Interface ATM5/1.200 address fe80::f7:0:231a:0
Administrative state enabled, Operational state enabled
Interface parameters:
    Version 2
    State No v1 Router Present
    Unsolicited report interval 100 (in 10ths of a second)
    5 multicast groups
Interface statistics:
    Rcvd: 0 v1 query, 0 v1 report, 25 v2 queries, 0 v2 report
    Sent: 0 v1 report, 0 v1 leave, 35 v2 reports
```

- Example 2

```
host1#show ipv6 mld-proxy interface brief

Interface       Intf Address       Ver State                UnSlTime
--------------- ------------------ --- -------------------- --------
ATM5/1.200      fe80::f7:0:231a:0    2 No v1 Router Present      100
```

Chapter 7

Configuring PIM for IPv6 Multicast

Protocol Independent Multicast (PIM) is a collection of multicast routing protocols that enable multicast routers to identify other multicast routers that can receive packets.

This chapter describes how to configure PIM for IPv6 multicast on the E-series router; it contains the following sections:

Overview

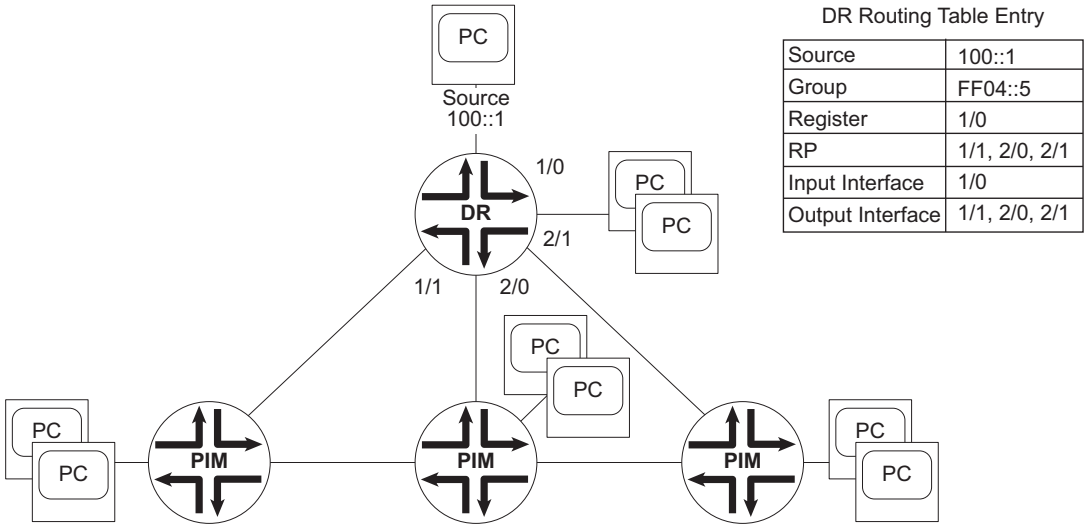
This implementation of PIM supports PIM sparse mode and PIM source-specific multicast (PIM SSM) for IPv6 multicast.

SSM is an extension to the Any Source Multicast (ASM) service model and facilitates the deployment of broadcast (one-to-many) applications, such as Internet TV and radio where large receiver audiences require traffic from a few well-known sources.

Figure 19 represents how PIM builds an (S,G) entry in an SRT. When multiple routers are connected to a multiaccess network, one router is assigned the role of the designated router. The designated router receives data from the source on interface 1/0 and multicasts the data to its downstream neighbors on interfaces 1/1, 2/0, and 2/1. In the designated router routing table, the entry for this operation lists the source as the IP address of the source and the group as the IP address of the multicast group.

Neighbors exchange hello messages periodically to determine the designated router. The router with the highest network layer address becomes the designated router. If the designated router subsequently receives a hello message from a neighbor with a higher network layer address, that neighbor becomes the designated router.

Figure 19: Source-Rooted Tree



PIM Sparse Mode

In addition to the features PIM sparse mode supports for IPv4, this IPv6 implementation of PIM sparse mode also supports remote neighbors.

For a description of PIM sparse mode, see *Chapter 3, Configuring PIM for IPv4 Multicast*.

Joining Groups

A host's designated router (DR) sends join messages to the RP when that host wants to join a group. When a host wants to leave a group, it communicates with its designated router through MLD. When the designated router no longer has any hosts that belong to a particular group, it sends a prune message to the RP.

Timers

PIM sparse mode uses timers to maintain the networking trees.

NOTE: PIM sparse mode routers poll their neighbors and hosts for various pieces of information at set intervals.

If a PIM sparse mode router does not receive information from a neighbor or host within a specific time, known as the *hold time*, it removes the associated information from its routing tables.

You can configure how often an interface sends hello messages (hello interval) and how often routers send RP announce messages (RP announce interval). The hold-time associated with hello messages is 3.5 times the hello interval, and the holdtime associated with RP announce messages is 2.5 times the RP announce interval.

All other timers are fixed and take the default values recommended in:

RFC 2934—Protocol Independent Multicast MIB for IPv4 (October 2000)

PIM Sparse Mode Bootstrap Router

PIM sparse mode routers need the address of the rendezvous point (RP) for each group for which they have (*,G) state. They obtain this address either through a bootstrap mechanism or through static configuration. Two bootstrap mechanisms exist—bootstrap router (BSR) or auto-RP. Auto-RP is not used in IPv6 implementations.

When implemented, BSR operates as follows:

1. One router in each PIM domain is elected the BSR.
2. All the routers in the domain that are configured to be RP candidates periodically unicast their candidacy to the BSR.
3. The BSR picks an RP set from the available candidates and periodically announces this set in a bootstrap message.
4. Bootstrap messages are flooded hop by hop throughout the domain until all routers in the domain learn the RP Set.

PIM Source-Specific Multicast

PIM source-specific multicast (SSM) is an extension of the PIM protocol. Using SSM, a client can receive multicast traffic directly from the source. PIM SSM uses PIM sparse mode functionality to create an SPT between the client and the source, but builds the SPT without using an RP.

By default, the SSM group multicast address is limited to the IPv6 address range FF3x::/96 where x represents any valid scope. You can use the **ipv6 pim ssm range** command to change the SSM group address range.

Advantages that an SSM-configured network has over a traditionally configured PIM sparse mode network include the following:

- No need for shared trees or RP mapping (no RP is required).
- No need for RP-to-RP source discovery through Multicast Source Discovery Protocol (MSDP).
- Simplified administrative deployment; you need only configure PIM sparse mode on all router interfaces and issue the necessary SSM commands (including specifying MLDv2 on the receiver local area network).
- Support for source lists; you can use source lists, supported in MLDv2, where only specified sources send traffic to the SSM group.

In a PIM SSM–configured network, the E-series router subscribes to an SSM channel (by means of MLDv2), announcing a desire to join group G and source S. The directly connected PIM sparse mode router, the designated router of the receiver, sends an (S,G) join message to its RPF neighbor for the source. For PIM SSM, the RP is not contacted in this process by the receiver (as happens in normal PIM sparse mode operations).

Platform Considerations

For information about modules that support PIM for IPv6 multicasting on the ERX-7xx models, ERX-14xx models, and the ERX-310 router:

- See *ERX Module Guide, Table 1, Module Combinations* for detailed module specifications.
- See *ERX Module Guide, Appendix A, Module Protocol Support* for information about the modules that support PIM for IPv6 multicasting.

For information about modules that support PIM for IPv6 multicasting on the E120 router and the E320 router:

- See *E120 and E320 Module Guide, Table 1, Modules and IOAs* for detailed module specifications.
- See *E120 and E320 Module Guide, Appendix A, IOA Protocol Support* for information about the modules that support PIM for IPv6 multicasting.

References

For more information about IPv6 multicast, see the following resources:

- RFC 2362—Protocol Independent Multicast-Sparse Mode (PIM-SM): Protocol Specification (June 1998)
- RFC 3569—An Overview of Source-Specific Multicast (SSM) (July 2003)
- Source-Specific Multicast for IP—draft-ietf-ssm-arch-06.txt (March 2005 expiration)
- Source-Specific Protocol Independent Multicast in 232/8—draft-ietf-mboned-ssm232-08.txt (September 2004 expiration)

Before You Begin

You can configure multicast on IPv4 and IPv6 interfaces.

For information about configuring IP interfaces, see *JUNOSe IP, IPv6, and IGP Configuration Guide, Chapter 1, Configuring IP*. For information about configuring IPv6 interfaces, see *JUNOSe IP, IPv6, and IGP Configuration Guide, Chapter 2, Configuring IPv6*.

For information about configuring PIM on IPv4 interfaces, see *Chapter 3, Configuring PIM for IPv4 Multicast*.

Enabling and Disabling PIM on a VR

By default, PIM is disabled. To enable PIM on a VR:

1. Enable multicast routing.
2. Create a VR, or access the VR context.
3. Create and enable PIM processing.

```
host1(config)#virtual-router boston
host1:boston(config)#ipv6 router pim
```

To disable PIM processing on a router, use the **pim disable** command.

ipv6 router pim

- Use to create and enable PIM processing on a VR or to access Router Configuration mode.
- Example

  ```
  host1:boston(config)#ipv6 router pim
  ```

- Use the **no** version to remove PIM from the VR.

pim disable

- Use to disable PIM processing. By default, PIM processing is enabled.
- Example

  host1:boston(config-router)#**pim disable**

- Use the **no** version to reenable PIM processing.

Enabling PIM on an Interface

You can enable PIM on an interface in one of the allowed modes and specify how often the interface sends hello messages to neighbors.

You can configure PIM and MLD on the same interface. If you configure MLD and PIM on an interface, the router considers that PIM owns the interface.

ipv6 pim query-interval

- Use to specify how often the router sends hello messages to neighbors.
- Example

  host1(config-if)#**ipv6 pim query-interval**

- Use the **no** version to restore the default setting, 30 seconds.

ipv6 pim sparse-mode

- Use to enable PIM in sparse mode on an interface.
- Example

  host1(config-if)#**ipv6 pim sparse-mode**

- Use the **no** version to disable PIM in sparse mode on an interface.

Configuring an RP Router for PIM Sparse Mode

When you use the router for PIM sparse mode, some VRs must act as RP routers. If you want to control PIM more tightly, you can configure a static RP router. To do so:

1. Configure an access list that details the multicast groups that can use the static RP router (in this case, all globally scoped multicast groups).

   host1(config)#**ipv6 access-list boston permit ff0e::/16 any**

2. Specify a static RP router.

   host1(config)#**ipv6 pim rp-address ::122:1 boston**

ipv6 pim rp-address

- Use to specify a static PIM RP router.
- Specify a standard IPv6 access list of multicast groups to control which multicast groups can use this RP router.
- Specify the **override** keyword if you want this static RP router to have priority over auto-RP routers.
- Example

  host1(config)#**ipv6 pim rp-address 2001::1 76 override**

- Use the **no** version to clear the filter from this interface.

Configuring BSR and RP Candidates for PIM Sparse Mode

When choosing candidate BSRs or candidate RPs, select well-connected routers in the core of the network.

Typically, candidate BSRs are a subset of the candidate RPs. A single BSR is elected for the domain the set of candidate BSRs. The elected BSR floods bootstrap messages (BSMs) containing their group-to-RP mappings to all PIM routers. PIM routers use the group-to-RP mappings supplied by the elected (or preferred) BSR.

Candidate RPs are routers that are capable of performing as a rendezvous point router for one or more multicast groups. Candidate RPs periodically advertise the set of groups they support to BSRs. A candidate RP may support all the multicast group address range or any subset thereof. You can achieve redundancy by configuring more than one candidate RP for a group or range of groups.

ipv6 pim bsr-candidate

- Use to define a router as a BSR candidate.
- To assign an interface from which the router should send messages, specify an interface type and specifier, such as atm 3/0. For details about interface types and specifiers, see *JUNOSe Command Reference Guide, About This Guide*.
- Specify a length (up to a 128 bits) for the hash mask length field sent in BSMs that the router originates. This mask is combined with the group address before the router calls the hash function. For example, specifying a value of 32 limits the group address to the first 32 bits. The default and maximum hash mask length is 126 bits.
- Use the **priority** keyword to specify a value for the BSR-priority field of BSMs that the router originates. In the BSR election process, the BSR with the higher priority is preferred. If the priority values are equal, the router with the higher IP address becomes the BSR. The default value is 0 (address comparison only).
- Use the **period** keyword to specify the interval (from 1 to 65535 seconds) at which the BSR sends bootstrap messages. The default value is 60 seconds.
- Example

  host1(config)#**ipv6 pim bsr-candidate loopback 1 30 10**

- Use the **no** version to stop the router from acting as a BSR candidate.

ipv6 pim rp-candidate

- Use to define a router as an RP router candidate.
- To assign an interface from which the router should send messages, specify an interface type and specifier, such as atm 3/0. For details about interface types and specifiers, see *JUNOSe Command Reference Guide, About This Guide*.
- Use the **group-list** keyword to specify an access list that contains the set of group prefixes supported by this candidate RP. If you do not specify a group list, the default is the entire multicast address range.

NOTE: You should not configure negative (that is, deny) access-list entries. BSR has no mechanism for distributing negative entries.

- Use the **hold-time** keyword to specify the amount of time the BSR keeps an RP in its candidate RP list if the BSR does not receive a candidate RP advertisement message. The default value is 150 seconds.
- Use the **priority** keyword to specify a priority field value that the candidate RP sends to the BSR in candidate RP advertisement messages. In the RP election process, the RP with the lower priority value is preferred. The default is 192.
- Use the **interval** keyword to specify an interval (from 1 to 65535 seconds) at which the candidate RP sends advertisement messages to the BSR. The default is 60 seconds.
- Example

  host1(config)#**access-list 1 permit 1001::1**
  host1(config)#**access-list 1 permit 1002::1**
  host1(config)#**ipv6 pim rp-candidate loopback 1 group-list 1**

- Use the **no** version to stop the router from acting as an RP candidate.

Switching to an SPT for PIM Sparse Mode

PIM sparse mode initiates multicast using a shared tree. You can configure PIM sparse mode to switch to an SPT when a source starts sending multicast messages, or you can prevent PIM sparse mode from switching to an SPT. Multicasting over an SPT can be more efficient than multicasting over a shared tree (see *PIM Sparse Mode* on page 80).

ipv6 pim spt-threshold

- Use to specify when PIM sparse mode switches from a shared tree to an SPT.
- Specify a nonzero integer or the keyword **infinity** to prevent PIM sparse mode from switching to an SPT.
- Specify a value of 0 to configure PIM to switch to an SPT when a source starts sending multicast messages.
- Example

  host1(config)#**ipv6 pim spt-threshold 4**

- Use the **no** version to restore the default, 0.

Configuring PIM Sparse Mode Remote Neighbors

You must use PIM sparse mode remote neighbors to run multicast services over BGP/MPLS VPNs.

NOTE: Although you can configure PIM sparse mode remote neighbors, you can not use these remote neighbors for BGP/MPLS VPNs.

To configure a pair of E-series routers to act as PIM remote neighbors:

1. On one router, specify the other router to be a remote neighbor, and identify the IP address of the interface on the other router that is used for the connection to this router.

   host1(config-router):boston#**remote-neighbor 1001::1 sparse-mode**

2. Specify the location of the local interface whose address is used as the source address for the PIM connection to a remote neighbor.

   host1(config-router-rn):boston#**update-source atm 2/1.108**

3. (Optional) Specify how often the router sends hello messages to the remote neighbor.

   host1(config-router-rn):boston#**query-interval 40**

4. Repeat Steps 2 to 3 for the other router.

query-interval

- Use to specify how often the router sends hello messages to remote neighbors.
- Example

  host1(config-router-rn)#**query-interval 40**

- Use the **no** version to restore the default setting, 30 seconds.

remote-neighbor

- Use to specify a remote neighbor for PIM sparse mode.
- Specify the IP address of the interface on the remote neighbor that PIM uses as the source address for the connection to this router.
- Example

  host1(config-router)#**remote-neighbor 1001::1 sparse-mode**

- Use the **no** version to remove the remote neighbor and any attributes configured for the remote neighbor.

update-source

- Use to specify the PIM interface whose local address is used as the source address for the PIM connection to a remote neighbor.
- You can use the same source address to form neighbor adjacencies with more than one PIM remote neighbor.
- You must use the IPv6 address of this interface when issuing the **remote-neighbor** command on the remote neighbor.
- Example

  host1(config-router-rn)#**update-source loopback 5**

- Use the **no** version to delete the source address from the connection to the remote neighbor.

Configuration Example

This example uses the configuration shown in Figure 19 on page 206. Two E-series routers called router Boston and router Chicago are running PIM and are connected by MPLS tunnels. To configure the routers as PIM remote neighbors:

1. Specify that router Chicago will be a remote neighbor of router Boston, and identify the IP address on router Chicago that will transmit datagrams to router Boston.

   boston(config-router)#**remote-neighbor 1001::1 sparse-mode**

2. Specify the location of the interface that will transmit datagrams from router Boston to router Chicago.

   boston(config-router-rn)#**update-source atm 2/1.108**

3. Specify that router Boston will send hello messages to router Chicago every 40 seconds.

   boston(config-if)#**ipv6 pim query-interval 40**

4. Specify that router Boston will be a remote neighbor of router Chicago, and identify the IP address on router Boston that will transmit datagrams to router Chicago.

   chicago(config-router)#**remote-neighbor 2001::1 sparse-mode**

5. Specify the location of the interface that will transmit datagrams from router Chicago to router Boston.

   chicago(config-router-rn)#**update-source atm 2/1.95**

6. Specify that router Chicago will send hello messages to router Boston every 40 seconds.

   chicago(config-if)#**ipv6 pim query-interval 40**

Using PIM Sparse Mode Join Filters

You can use PIM sparse mode join filters to prevent multicast state from being created in the PIM sparse mode router. The filters are applied to join entries in PIM join/prune messages that are received from PIM sparse mode neighbors.

By denying joins at the edge of a network, you can limit the multicast state and traffic in the network. By accepting only certain joins, you can control which multicast services an end user can receive. PIM join filters also reduce the potential for denial of service (DOS) attacks where large numbers of joins forwarded to each router on the RPT can result in a PIM state explosion and very high memory consumption.

For information about how to create access lists, see *JUNOSe IP Services Configuration Guide, Chapter 1, Configuring Routing Policy*.

ipv6 pim join-filter

- Use to specify an extended access list that you want this PIM interface to use as a join filter.
- You can apply the join filter at the global level or at the interface level.
- If an interface-level filter exists, it takes precedence over the global-level filter.
- Example 1

  host1(config)#**ipv6 pim join-filter gold**

- Example 2

  host1(config-interface)#**ipv6 pim join-filter gold**

- Use the **no** version to remove the filter association.

Configuring PIM SSM

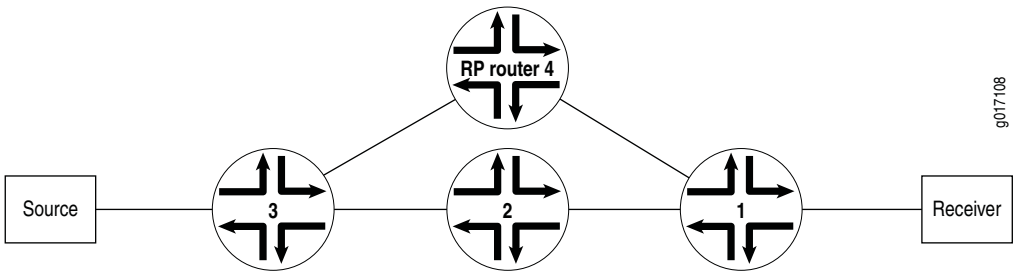
Source Specific Multicast (SSM) is a datagram delivery model that best supports one-to-many applications, also known as broadcast applications. SSM is networking technology that targets audio and video broadcast application environments.

To use PIM SSM, MLDv2 must be configured on customer premises equipment (CPE)–facing interfaces to receivers, and PIM sparse mode must be configured on CPE-facing interfaces to sources and on core-facing interfaces. After configuring SSM, you can use the **show ipv6 pim sparse-mode sg-state** command to display SSM group membership information.

To configure PIM SSM, you enable PIM SSM on the router and define the SSM range of IP multicast addresses.

Figure 20 shows how PIM SSM is configured between a receiver and a source in the network. Interface 1 has MLDv2 enabled and all other interfaces towards the core or source have PIM SSM enabled.

Figure 20: Network on Which to Configure PIM SSM



To configure PIM SSM:

1. Enable PIM SSM on the E-series router. The IANA SSM range is configured by default. You can modify the SSM address range by using the access list.

   host1(config)#**access-list 15 permit ip any host 239.0.0.2**
   host1(config)#**access-list 15 permit ip any 232.0.0.0 0.225.225.225**
   host1(config)#**ipv6 pim ssm range 15**

2. Enable PIM sparse mode on the CPE-facing interface towards the source or core.

3. Enable MLDv2 on the CPE-facing interface towards the receiver.

PIM SSM also works with MLDv1 if you configure the ssm-map in MLD as in the following example:

host1(config)#**ipv6 pim ssm**
host1(config)#**ipv6 access-list ssm_map1 permit any host ff3e::1**
host1(config)#**ipv6 mld ssm-map enable**
host1(config)#**ipv6 mld ssm-map static ssm_map1 51::1**

The **no** version disables ssm-map:

host1(config)#**no ipv6 mld ssm-map static ssm_map1 51::1**

ipv6 pim ssm

- Use to enable PIM SSM and define the SSM range of IPv6 multicast addresses.
- Example 1—Enables SSM with addresses in the IANA range. The SSM address range is set as the default, which is limited to the IPv6 address range, and where *x* represents any valid scope.

  host1(config)#**ipv6 pim ssm FF3x::/96**

- Example 2—Configures Class D addresses outside of the default SSM range.

  host1(config)#**ipv6 access-list alist permit any ff3e::1:0:0/96**
  host1(config)#**ipv6 pim ssm range alist**

- Example 3—Resets the SSM address range to the default.

  host1(config)#**ipv6 pim ssm default**

- Use the **no** version to disable SSM.

Configuring the BFD Protocol for PIM

The **ipv6 pim bfd-liveness-detection** command configures the Bidirectional Forwarding Detection (BFD) protocol for PIM. The BFD protocol uses control packets and shorter detection time limits to more rapidly detect failures in a network. Also, because they are adjustable, you can modify the BFD timers for more or less aggressive failure detection.

PIM routers send periodic hello messages from each PIM-enabled interface. You can configure this interval using the **ipv6 pim query-interval** command. By default, the PIM router sends a hello message every 30 seconds (with an interval range of 0–210 seconds). If it receives no response from a neighbor within 3.5 times the interval value (a minimum of 3.5 seconds), the PIM router drops the neighbor.

In contrast, when a BFD session exists between neighbors, a PIM neighbor that goes down is detected quickly (in milliseconds rather than in seconds).

When you issue the **ipv6 pim bfd-liveness-detection** command on a PIM router, the router establishes BFD liveness detection with all BFD-enabled PIM neighbors. When the local router receives an update from a remote PIM neighbor—if BFD is enabled and if the session is not already present—the local router attempts to create a BFD session to the remote neighbor.

Each adjacent pair of neighbors negotiates an acceptable transmit interval for BFD packets. The negotiated value can be different on each neighbor. Each neighbor then calculates a BFD liveness detection interval. When a neighbor does not receive a BFD packet within the detection interval, it declares the BFD session to be down.

NOTE: Before the router can use the **ipv6 pim bfd-liveness-detection** command, you must specify a BFD license key. To view an already configured license, use the **show license bfd** command.

For general information about configuring and monitoring the BFD protocol, see *JUNOSe IP Services Configuration Guide, Chapter 5, Configuring BFD*.

ipv6 pim bfd-liveness-detection

- Use to enable BFD (bidirectional forwarding detection) and define BFD values to more quickly detect PIM data path failures.
- The neighbors in a PIM network use the configured values to negotiate the actual transmit intervals for BFD packets.
  - You can use the **minimum-transmit-interval** keyword to specify the interval at which the local router proposes to transmit BFD control packets to its neighbors. The default value is 300 milliseconds.
  - You can use the **minimum-receive-interval** keyword to specify the minimum interval at which the local router must receive BFD control packets from its neighbors. The default value is 300 milliseconds.
  - You can use the **minimum-interval** keyword to specify the same value for both of those intervals. Configuring a minimum interval has the same effect as configuring the minimum receive interval and the minimum transmit interval to the same value. The default value is 300 milliseconds.

- You can use the **multiplier** keyword to specify the detection multiplier value. The calculated BFD liveness detection interval can be different on each neighbor. The multiplier value is roughly equivalent to the number of packets that can be missed before the BFD session is declared to be down. The default value is 3.
- For details on liveness detection negotiation, see *Negotiation of the BFD Liveness Detection Interval* section in *JUNOSe IP Services Configuration Guide, Chapter 5, Configuring BFD*.
- You can change the BFD liveness detection parameters at any time without stopping or restarting the existing session; BFD automatically adjusts to the new parameter value. However, no changes to BFD parameters take place until the values resynchronize with each neighbor.
- Example

  host1(configif)#**ipv6 pim bfd-liveness-detection minimum-interval 800**

- Use the **no** version to disable BFD on the PIM interface.

Removing PIM

To remove PIM from a VR, use the **no ipv6 router pim** command.

ipv6 router pim

- Use to create and enable PIM processing on a VR or to access Router Configuration mode.
- Example

  host1:boston(config)#**no ipv6 router pim**

- Use the **no** version to remove PIM from the VR.

Resetting PIM Counters and Mappings

You can use the **clear ipv6 pim** commands to reset PIM counters and mappings.

clear ipv6 pim interface

- Use to clear the counters for multicast packet statistics on all interfaces or a specified interface.
- Specify an interface type and identifier, such as atm 3/0 to clear the counters on that interface. For details about interface types and specifiers, see *JUNOSe Command Reference Guide, About This Guide*.
- If you do not specify an interface, the router clears the counters on all interfaces.
- Example

  host1#**clear ipv6 pim interface atm 3/0.5 count**

- There is no **no** version.

clear ipv6 pim remote-neighbor

- Use to clear the counters for remote neighbor statistics on all interfaces or the specified interface.
- Specify the IP address of an interface to clear the counters for that interface.
- If you do not specify an interface, the router clears the counters on all interfaces.
- Example

  host1#**clear ipv6 pim remote-neighbor 1001::1 count**

- There is no **no** version.

Monitoring PIM

You can display information about PIM events and parameters.

Monitoring PIM Events

You can use the debug PIM commands to view information about PIM events.

debug ipv6 pim

- Use to show information about the selected event.
- To control the type of events displayed, specify a severity level.
- To control how much information to display, specify a verbosity level.
- Example

  host1#**debug ipv6 pim events severity 1 verbosity low**

- Use the **no** version to disable the display.

undebug ipv6 pim

- Use to turn off the display of information previously enabled with the **debug ipv6 pim** command.
- Example

  host1#**undebug ipv6 pim events**

- There is no **no** version.

Monitoring PIM Settings

You can use the **show ipv6 pim** commands to display information about PIM settings.

show ipv6 pim

- Use to view general PIM router-level information.
- Field descriptions
  - Default PIM Version—Default PIM version number (always 2)
  - Default Domain Id—Default Domain Id (always 0)
  - Default Hello period—Default interval (in minutes) at which the router sends hello messages to neighbors
  - Default Hello Hold Time—Default time (in minutes) for which the router keeps the neighbor state alive
  - Default J / P Hold Time—Hold time value (in seconds) set in Join/Prune messages originated by this PIM router
  - Keepalive Period—Time SG join state is maintained in the absence of SG Join message
  - Assert Time—Period after last assert before assert state is timed out
  - Register Suppression Time—Period during which a designated router stops sending registers to the RP
  - Register Probe Time—Time before register suppression time (RST) expires when a designated router might send a NULL-Register to the RP
  - Register TTL—TTL value (in PIM register packets) originated by this PIM router
  - SSM—State of SSM on this PIM router (enabled or disabled)
  - range—Default SSM group range or name of the access list specifying the range
  - Join filter—Name of the join filter access-list (if configured) for this PIM router
- Example

```
host1:1#show ipv6 pim
Default PIM Version: 2
Default Domain Id: 0
Default Hello Period: 30
Default Hello HoldTime: 105
Default J/P HoldTime: 210
Keepalive Period: 210
Assert Time: 210
Register Suppression Time: 60
Register Probe Time: 5
Register TTL: 64
SSM enabled, range default
Join filter, access-list bronze
```

show ipv6 pim bsr

- Use to display BSR information and the group prefixes for which the local router is a candidate RP in a PIM sparse mode environment.
- Field descriptions
  - Candidacy—Whether or not the router is a candidate BSR
  - Configured on—Interface on which the router is configured
  - address—Address of the router
  - hashMaskLen—Hash mask length
  - priority—Priority of the router
  - period—Time between bootstrap messages
  - Elected BSR—"this router" or IP address of the elected bootstrap router
  - next BSM—If BSR is "this router," time until the next bootstrap message is sent
  - expires in—If BSR is not "this router," time until the elected BSR expires if no bootstrap messages are received
  - Local candidate RP mapping(s)—Routers that the mapping agent is evaluating to determine an RP router for this interface
- Example 1—On a router that is the elected BSR

```
host1:1#show ipv6 pim bsr

This PIM router is a Candidate BSR.
  Configured on intf ATM3/0.101, address: ::107:9
  hashMaskLen 30, priority 2, period 60 seconds.
Elected BSR is this router, next BSM in 3 seconds.

Local candidate RP mapping(s):
Candidate RP ::107:9
::108:86, BSR, hold-time 150, interval 60, priority 192
::108:87, BSR, hold-time 150, interval 60, priority 192, from access-list acl
::108:88, BSR, hold-time 150, interval 60, priority 192, from access-list acl
```

- Example 2—On a router that is a candidate BSR

```
host1:1#show ipv6 pim bsr
This PIM router is a Candidate BSR.
  Configured on intf ATM3/0.100, address: ::107:9
  hashMaskLen 30, priority 2, period 60 seconds.
Elected BSR is ::107:8 (priority 0), expires in 73 seconds.
```

- Example 3—On a router that is not a candidate BSR

```
host1:1#show ipv6 pim bsr
This PIM router is not a Candidate BSR.
Elected BSR is ::107:9 (priority 0), expires in 73 seconds.
```

show ipv6 pim interface

- Use to display information about PIM interfaces.
- Specify no keywords or variables to view information about all PIM interfaces.
- Provide an interface type and specifier (such as atm 3/0) to display information about that interface only. For details about interface types and specifiers, see *JUNOSe Command Reference Guide, About This Guide*.
- Specify the **detail** keyword to view detailed information for all PIM interfaces or for a specified PIM interface.
- Specify the **summary** keyword to view the number of configured, enabled, and disabled PIM sparse-mode interfaces.
- Specify the **count** keyword to view the number of multicast packets that the interface has sent and received.
- Field descriptions
  - Interface Addr—IPv6 address of the interface
  - Interface Name—Type and identifier of the interface. For details about interface types and specifiers, see *JUNOSe Command Reference Guide, About This Guide*.
  - Ver—Version of PIM running on this interface
  - Mode—PIM mode running on this interface: Sparse
  - Nbr Count—Number of neighbors connected to this interface
  - Hello Intvl—Time interval at which the interface sends hello messages to neighbors
  - DR Address—Address of the designated router
  - SM—Number of PIM sparse mode interfaces
    - enabled—Number of interfaces administratively enabled
    - disabled—Number of interfaces administratively disabled
  - ControlPkt Count In | Out—PIM messages received on and sent from this interface
    - Hello—Number of hello messages
    - JoinPrune—Number of join and prune messages
    - Assert—Number of assert messages
- Example 1

```
host1#show ipv6 pim interface
PIM Interface Table
Interface Addr  Interface Name        Ver Mode        Nbr   Hello  DR Addr
                                                      Count Intvl
::108:85        atm2/1.108            2   Sparse      1     30     ::108:85
::107:84        atm2/1.109            2   Sparse      1     30     ::107:84
::111:89        atm2/0.110            2   Sparse      1     30     ::111:89
::110:8c        loopback8             2   Sparse      0     30     ::110:8c
```

- Example 2

```
host1#show ipv6 pim interface summary
PIM Interface Summary
  SM:    0, 0 enabled, 0 disabled
```

- Example 3

```
host1#show ipv6 pim interface count
PIM Interface Count
Interface Addr  Interface Name       ControlPktCount In|Out
                                     Hello     JoinPrune  Assert
::107:84        ATM3/0.20             0         0          0
                                      0         0          0
```

show ipv6 pim neighbor

- Use to display information about PIM neighbors that the router discovered.
- Provide an interface type and specifier (such as atm 3/0) to display information about that interface only. For details about interface types and specifiers, see *JUNOSe Command Reference Guide, About This Guide*.
- Specify the **detail** keyword to view detailed information for all PIM neighbors or for a specified PIM neighbor.
- Field descriptions
  - Neighbor Addr—IPv6 address of the neighbor
  - Interface Name—Type and specifier of the interface to which the neighbor connects. For details about interface types and specifiers, see *JUNOSe Command Reference Guide, About This Guide*.
  - Uptime—Time since the router discovered this neighbor
  - Expires—Time available for the neighbor to send a hello message to the interface. If the neighbor does not send a hello message during this time, it will no longer be a neighbor.
  - Ver—Version of PIM that the neighbor is running
  - Mode—PIM mode that the neighbor is using: sparse
- Example

```
host1#show ipv6 pim neighbor
PIM Neighbor Table
Neighbor Addr   Interface Name        Uptime         Expires  Ver Mode
::107:48        atm2/1.109            1d15:47:35     00:01:41 2   Sparse
::108:58        atm2/1.108            1d15:47:34     00:01:42 2   Sparse
::111:98        atm2/0.110            1d15:48:02     00:01:44 2   Sparse
```

show ipv6 pim remote-neighbor

- Use to view information about PIM remote neighbors.
- Field descriptions
  - Remote Nbr Addr—IPv6 address of remote neighbor
  - OurEnd Addr—IPv6 address of local interface, such as the local endpoint of a tunnel, that transmits data to remote neighbor
  - Ver—Version of PIM running on the local interface
  - Mode—PIM mode running on the local interface; always PIM sparse mode
  - Nbr Count—Number of remote neighbors detected: 0 or 1
  - Hello Intvl—Time interval at which the interface sends hello messages to neighbors
  - DR Addr—Address of designated router
  - In interface—Type and identifier of the interface on which PIM router receives packets from remote neighbor. For details about interface types and specifiers, see *JUNOSe Command Reference Guide, About This Guide*.
  - Out interface—Type and identifier of the interface on which PIM router sends packets to remote neighbor. For details about interface types and specifiers, see *JUNOSe Command Reference Guide, About This Guide*.
- Example

```
host1:boston#show ipv6 pim remote-neighbor
PIM RemoteNbr Table
RemoteNbr Addr  OurEnd Addr     Ver Mode        Nbr   Hello  DR Addr
                                                Count Intvl
1001::1         2001::1         2   Sparse      1     30     ::107:84
      In interface : atm2/1.109
      Out interface: atm2/1.108
```

show ipv6 pim rp

- Use to display information about PIM group-to-RP mappings.
- Specify the address of a group to view PIM group-to-RP mappings for a particular group.
- To display all group-to-RP mappings that the router has recorded, specify the **mapping** keyword.
- Field descriptions
  - Group—Prefix of the multicast group
  - RP—IP address of RP router for the multicast group
  - priority—This field is not functional
  - via—Method by which the RP router was assigned (static, BSR)
- Example

```
host1:8#show ipv6 pim rp mapping
PIM Group-to-RP mapping(s)
Group(s) ff00::/12
  RP ::122:1,  priority 0,  via  static
Group(s) ff0e::1:0/96
  RP ::120:1,  priority 0,  via  static
```

show ipv6 pim rp-hash

- Use to show which RP router a multicast group is using.
- Field descriptions
  - Group—Multicast group
  - RP—RP router for the multicast group
  - priority—This field is not functional
  - via—Method by which the RP router was assigned (static, BSR)
- Example

```
host1:2#show ipv6 pim rp-hash 232.1.1.1
Group(s) ff00::/12
  RP ::122:1,  priority 0,  via  static
```

show ipv6 pim sparse-mode sg-state

- Use to display information for each (S,G) entry for PIM sparse mode and PIM SSM.
- Field descriptions
  - Group-to-RP mapping—IPv6 addresses and network mask of the multicast group
  - RP—IPv6 address of RP router
  - SSM group—Indicates that this is an SSM group
  - RPF route—IPv6 address and network mask of the RPF route
  - IIF—IPv6 address of the incoming interface for the RPF route
  - UpNbr—IPv6 address of the upstream neighbor
  - Oifs—Outgoing interface
  - Register Oif to RP—IP address of RP router for the outgoing interface; suppressed for SSM
  - Address—IPv6 address of outgoing interface
  - Interface—Type and specifier of the interface. For details about interface types and specifiers, see *JUNOSe Command Reference Guide, About This Guide*.
  - Joined as—Type of mapping
    - (S,G)—Mapping from a specific source to a specific group
    - (*,G)—Mapping from any source to a specific group
    - (*,*,RP)—Mapping from any source to any group
  - Join expires—Number of seconds before the (S,G) membership expires
  - Count of entries—Total count of (S,G) pair mappings
- Example

```
host1:2#show ipv6 pim sparse-mode sg-state
PIM SM route table and oif information
<*, ff0e::1:3>
  Group-to-RP mapping: ff00::/12 RP: ::123:1
  RPF Route: ::123:1/96 IIF: :106:73 UpNbr: ::106:37
```

```
  Oifs:
   Address: ::78:7:7 Interface: loopback7
    Local group membership present.
<*, ff0e::a:1>
  Group-to-RP mapping: ff001:/12 RP: ::123:1
  RPF Route: :123:1/96 IIF: :106:73 UpNbr: :106:37
  Oifs:
   Address: ::78:7:7 Interface: loopback7
    Local group membership present.

<::118:34, ff3e::1>
  SSM Group
  RPF Route: ::118:0/96 IIF: :118:1 (Directly attached)
  Oifs:
   Register Oif to RP: ::141:2 suppressed for SSM Group.
   Address: ::134:1 Interface: ATM3/0.104
    Joined as <S, G>   Join Expires: 161

<::118:35, ff3e::1>
  SSM Group
  RPF Route: ::118:0/96 IIF: ::118:1 (Directly attached)
  Oifs:
   Register Oif to RP: ::141:2 suppressed for SSM Group.
   Address: ::134:1 Interface: ATM3/0.104
    Joined as <S, G>   Join Expires: 161

<::10:8, ff0e::5:1>      EntryExpires: 143
  Group-to-RP mapping: ff00::/12 RP: ::123:1
  RPF Route: ::10:0/96 IIF: ::106:73 UpNbr: ::106:37
  Oifs:
   Address: ::78:7:7 Interface: loopback7
    Joined as <*, G>

Count of entries - <S, G>    : 3
                   <*, G>    : 2
                   <*, *, RP>: 0
```

show ipv6 pim sparse-mode unicast-route

- Use to display the unicast routes that PIM sparse mode is using.
- Field descriptions
  - Route—IPv6 address and network mask for the unicast route
  - RpfNbr—RPF neighbor
  - Iif—Incoming interface for the unicast route
  - Pref—Preference for the unicast route
  - Metric—Value of metric for the unicast route (type of metric varies with the unicast protocol)
  - Count of entries—Number of unicast routes that PIM sparse mode is using.
- Example

```
host1:2#show ipv6 pim sparse-mode unicast-route
PIM SM unicast route table information
Route                          RpfNbr          Iif             Pref  Metric
------------------------------------------------------------------------------
::122:0       /96                              ::122:1         255   1
Count of entries: 1
```

show ipv6 pim spt-threshold

- Use to display the threshold for switching to the shortest path tree at a PIM designated router.
- Field descriptions
  - Access List Name—Name of the IPv6 access list that specifies the groups to which the threshold applies
  - SptThreshold (in kbps)—Value at which PIM sparse mode should switch from a shared tree to an SPT. A value of infinity indicates that PIM sparse mode should never switch to an SPT.
- Example

```
host1:2#show ipv6 pim spt-threshold
Access List Name                 SptThreshold(in kbps)
-------------------------------------------------------
1                                infinity
```

Index

F

G

H

I

J

L

M

N

P

Q

R

S

T

U

V