



IP Multicast Routing Configuration Guide, Cisco IOS XE Cupertino 17.9.x (Catalyst 9200 Switches)

First Published: 2022-08-01

Americas Headquarters

Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
<http://www.cisco.com>
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 527-0883

THE SPECIFICATIONS AND INFORMATION REGARDING THE PRODUCTS IN THIS MANUAL ARE SUBJECT TO CHANGE WITHOUT NOTICE. ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS MANUAL ARE BELIEVED TO BE ACCURATE BUT ARE PRESENTED WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. USERS MUST TAKE FULL RESPONSIBILITY FOR THEIR APPLICATION OF ANY PRODUCTS.

THE SOFTWARE LICENSE AND LIMITED WARRANTY FOR THE ACCOMPANYING PRODUCT ARE SET FORTH IN THE INFORMATION PACKET THAT SHIPPED WITH THE PRODUCT AND ARE INCORPORATED HEREIN BY THIS REFERENCE. IF YOU ARE UNABLE TO LOCATE THE SOFTWARE LICENSE OR LIMITED WARRANTY, CONTACT YOUR CISCO REPRESENTATIVE FOR A COPY.

The Cisco implementation of TCP header compression is an adaptation of a program developed by the University of California, Berkeley (UCB) as part of UCB's public domain version of the UNIX operating system. All rights reserved. Copyright © 1981, Regents of the University of California.

NOTWITHSTANDING ANY OTHER WARRANTY HEREIN, ALL DOCUMENT FILES AND SOFTWARE OF THESE SUPPLIERS ARE PROVIDED "AS IS" WITH ALL FAULTS. CISCO AND THE ABOVE-NAMED SUPPLIERS DISCLAIM ALL WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, THOSE OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NON-INFRINGEMENT OR ARISING FROM A COURSE OF DEALING, USAGE, OR TRADE PRACTICE.

IN NO EVENT SHALL CISCO OR ITS SUPPLIERS BE LIABLE FOR ANY INDIRECT, SPECIAL, CONSEQUENTIAL, OR INCIDENTAL DAMAGES, INCLUDING, WITHOUT LIMITATION, LOST PROFITS OR LOSS OR DAMAGE TO DATA ARISING OUT OF THE USE OR INABILITY TO USE THIS MANUAL, EVEN IF CISCO OR ITS SUPPLIERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.

All printed copies and duplicate soft copies of this document are considered uncontrolled. See the current online version for the latest version.

Cisco has more than 200 offices worldwide. Addresses and phone numbers are listed on the Cisco website at www.cisco.com/go/offices.

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: <https://www.cisco.com/c/en/us/about/legal/trademarks.html>. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1721R)

© 2022 Cisco Systems, Inc. All rights reserved.



CONTENTS

CHAPTER 1

| | |
|--|----------|
| IP Multicast Routing Technology Overview | 1 |
| Information About IP Multicast Technology | 1 |
| About IP Multicast | 1 |
| Role of IP Multicast in Information Delivery | 2 |
| Multicast Group Transmission Scheme | 2 |
| IP Multicast Routing Protocols | 4 |
| Internet Group Management Protocol | 5 |
| Protocol-Independent Multicast | 5 |
| Rendezvous Point | 5 |
| IGMP Snooping | 5 |
| IP Multicast Tables | 6 |
| Hardware and Software Forwarding | 7 |
| Partial Routes | 8 |
| Software Routes | 8 |
| Non-Reverse Path Forwarding Traffic | 8 |
| IP Multicast Boundary | 9 |
| IP Multicast Group Addressing | 10 |
| IP Class D Addresses | 10 |
| IP Multicast Address Scoping | 10 |
| Layer 2 Multicast Addresses | 12 |
| Cisco Express Forwarding, MFIB, and Layer 2 Forwarding | 12 |
| IP Multicast Delivery Modes | 14 |
| Source Specific Multicast | 14 |
| Multicast Fast Drop | 14 |
| Multicast Forwarding Information Base | 15 |
| S/M, 224/4 | 16 |

Multicast High Availability 16
 Additional References for IP Multicast 16
 Feature History for IP Multicast Routing Technology 19

CHAPTER 2

Configuring Basic IP Multicast Routing 21

Information About Basic IP Multicast Routing 21
 Multicast Forwarding Information Base Overview 21
 Default IP Multicast Routing Configuration 22
 How to Configure Basic IP Multicast Routing 22
 Configuring Basic IP Multicast Routing 22
 Configuring IP Multicast Forwarding 24
 Configuring a Static Multicast Route (mroute) 25
 Configuring Multicast VRFs 27
 Configuring Optional IP Multicast Routing Features 28
 Defining the IP Multicast Boundary 29
 Configuring sdr Listener Support 30
 Monitoring and Maintaining Basic IP Multicast Routing 32
 Clearing Caches, Tables, and Databases 32
 Displaying System and Network Statistics 32
 Configuration Examples for Basic IP Multicast Routing 34
 Example: Configuring an IP Multicast Boundary 34
 Example: Responding to mroute Requests 34
 Additional References for Basic IP Multicast Routing 35
 Feature History for Basic IP Multicast Routing 35

CHAPTER 3

Configuring Multicast Routing over GRE Tunnel 37

Prerequisites for Configuring Multicast Routing over GRE Tunnel 37
 Restrictions for Configuring Multicast Routing over GRE Tunnel 37
 Information About Multicast Routing over GRE Tunnel 38
 How to Configure Multicast Routing over GRE Tunnel 38
 Configuring a GRE Tunnel to Connect Non-IP Multicast Areas 38
 Tunneling to Connect Non-IP Multicast Areas Example 39
 Additional References for Multicast Routing over GRE Tunnel 41
 Feature History for Multicast Routing over GRE Tunnel 41

CHAPTER 4**Configuring IGMP 43**

| | |
|---|----|
| Prerequisites for IGMP and IGMP Snooping | 43 |
| Prerequisites for IGMP Snooping | 43 |
| Restrictions for IGMP and IGMP Snooping | 44 |
| Restrictions for Configuring IGMP | 44 |
| Restrictions for IGMP Snooping | 44 |
| Information about IGMP | 45 |
| Role of the Internet Group Management Protocol | 45 |
| IGMP Multicast Addresses | 45 |
| IGMP Versions | 45 |
| IGMP Version 1 | 46 |
| IGMP Version 2 | 46 |
| IGMP Version 3 | 46 |
| IGMPv3 Host Signaling | 46 |
| IGMP Versions Differences | 46 |
| IGMP Join and Leave Process | 49 |
| IGMP Join Process | 49 |
| IGMP Leave Process | 50 |
| IGMP Snooping | 50 |
| Joining a Multicast Group | 51 |
| Leaving a Multicast Group | 53 |
| Immediate Leave | 54 |
| IGMP Configurable-Leave Timer | 54 |
| IGMP Report Suppression | 54 |
| IGMP Snooping and Device Stacks | 54 |
| IGMP Filtering and Throttling | 55 |
| Default IGMP Configuration | 55 |
| Default IGMP Snooping Configuration | 56 |
| Default IGMP Filtering and Throttling Configuration | 56 |
| How to Configure IGMP | 57 |
| Configuring the Device as a Member of a Group | 57 |
| Changing the IGMP Version | 58 |
| Modifying the IGMP Host-Query Message Interval | 59 |

| | |
|---|----|
| Changing the Maximum Query Response Time for IGMPv2 | 61 |
| Configuring the Device as a Statically Connected Member | 62 |
| Configuring IGMP Profiles | 63 |
| Applying IGMP Profiles | 65 |
| Setting the Maximum Number of IGMP Groups | 66 |
| Configuring the IGMP Throttling Action | 67 |
| Configuring the Device to Forward Multicast Traffic in the Absence of Directly Connected IGMP Hosts | 69 |
| Controlling Access to an SSM Network Using IGMP Extended Access Lists | 70 |
| How to Configure IGMP Snooping | 72 |
| Enabling IGMP Snooping | 72 |
| Enabling or Disabling IGMP Snooping on a VLAN Interface | 73 |
| Setting the Snooping Method | 74 |
| Configuring a Multicast Router Port | 75 |
| Configuring a Host Statically to Join a Group | 76 |
| Enabling IGMP Immediate Leave | 77 |
| Configuring the IGMP Leave Timer | 78 |
| Configuring the IGMP Robustness-Variable | 80 |
| Configuring the IGMP Last Member Query Count | 81 |
| Configuring TCN-Related Commands | 82 |
| Controlling the Multicast Flooding Time After a TCN Event | 82 |
| Recovering from Flood Mode | 83 |
| Disabling Multicast Flooding During a TCN Event | 84 |
| Configuring the IGMP Snooping Querier | 85 |
| Disabling IGMP Report Suppression | 87 |
| Monitoring IGMP | 88 |
| Monitoring IGMP Snooping Information | 89 |
| Monitoring IGMP Filtering and Throttling Configuration | 90 |
| Configuration Examples for IGMP | 91 |
| Example: Configuring the Device as a Member of a Multicast Group | 91 |
| Example: Controlling Access to Multicast Groups | 91 |
| Examples: Configuring IGMP Snooping | 91 |
| Example: Configuring IGMP Profiles | 92 |
| Example: Applying IGMP Profile | 92 |

| | |
|--|----|
| Example: Setting the Maximum Number of IGMP Groups | 93 |
| Example: Interface Configuration as a Routed Port | 93 |
| Example: Interface Configuration as an SVI | 93 |
| Example: Configuring the Device to Forward Multicast Traffic in the Absence of Directly Connected IGMP Hosts | 94 |
| Controlling Access to an SSM Network Using IGMP Extended Access Lists | 94 |
| Example: Denying All States for a Group G | 94 |
| Example: Denying All States for a Source S | 95 |
| Example: Permitting All States for a Group G | 95 |
| Example: Permitting All States for a Source S | 95 |
| Example: Filtering a Source S for a Group G | 95 |
| Additional References for IGMP | 96 |
| Feature History for IGMP | 96 |

CHAPTER 5**Configuring IGMP Proxy 97**

| | |
|--|-----|
| Prerequisites for IGMP Proxy | 97 |
| Information About IGMP Proxy | 97 |
| IGMP Proxy | 97 |
| IGMP Proxy for a Single Upstream Interface | 97 |
| IGMP Proxy for Multiple Upstream Interfaces | 100 |
| How to Configure IGMP Proxy | 101 |
| Configuring the Upstream UDL Device for IGMP UDLR | 101 |
| Configuring the Downstream UDL Device for IGMP UDLR with IGMP Proxy Support | 102 |
| Configuring the Downstream Device for IGMP Proxy Join without UDLR | 104 |
| Configuring the Downstream Device for IGMP Proxy for Multiple Upstream Interfaces | 106 |
| Configuration Examples for IGMP Proxy | 107 |
| Example: Configuring the Upstream UDL Device for IGMP UDLR | 107 |
| Example: Configuring the Downstream UDL Device for IGMP UDLR with IGMP Proxy Support | 107 |
| Example: Configuring the Downstream Device for IGMP Proxy Join without UDLR | 108 |
| Example: Configuring the Downstream Device for IGMP Proxy for Multiple Upstream Interfaces | 108 |
| Additional References for IGMP Proxy | 108 |
| Feature History for IGMP Proxy | 109 |

CHAPTER 6**Constraining IP Multicast in Switched Ethernet 111**

| | |
|---|-----|
| Prerequisites for Constraining IP Multicast in a Switched Ethernet Network | 111 |
| Information About IP Multicast in a Switched Ethernet Network | 111 |
| IP Multicast Traffic and Layer 2 Switches | 111 |
| CGMP on Catalyst Switches for IP Multicast | 112 |
| IGMP Snooping | 112 |
| Router-Port Group Management Protocol (RGMP) | 112 |
| How to Constrain Multicast in a Switched Ethernet Network | 113 |
| Configuring Switches for IP Multicast | 113 |
| Configuring IGMP Snooping | 113 |
| Enabling CGMP | 113 |
| Configuring IP Multicast in a Layer 2 Switched Ethernet Network | 114 |
| Configuration Examples for Constraining IP Multicast in a Switched Ethernet Network | 115 |
| RGMP Configuration Example | 115 |
| Additional References for Constraining IP Multicast in a Switched Ethernet Network | 116 |
| Feature History for Constraining IP Multicast in Switched Ethernet | 116 |

CHAPTER 7
Configuring Protocol Independent Multicast (PIM) 119

| | |
|---|-----|
| Prerequisites for PIM | 119 |
| Restrictions for PIM | 120 |
| PIMv1 and PIMv2 Interoperability | 120 |
| Restrictions for Configuring PIM Stub Routing | 120 |
| Restrictions for Configuring Auto-RP and BSR | 121 |
| Restrictions for Auto-RP Enhancement | 122 |
| Information about PIM | 122 |
| Protocol Independent Multicast Overview | 122 |
| PIM Versions | 122 |
| Multicast Source Discovery Protocol (MSDP) | 123 |
| PIM Sparse Mode | 123 |
| PIM Stub Routing | 124 |
| Rendezvous Points | 125 |
| Auto-RP | 125 |
| The Role of Auto-RP in a PIM Network | 126 |
| Multicast Boundaries | 126 |
| Sparse-Dense Mode for Auto-RP | 127 |

| | |
|---|-----|
| Auto RP Benefits | 128 |
| PIM Domain Border | 128 |
| PIMv2 Bootstrap Router | 128 |
| Multicast Forwarding | 129 |
| Multicast Distribution Source Tree | 129 |
| Multicast Distribution Shared Tree | 130 |
| Source Tree Advantage | 131 |
| Shared Tree Advantage | 131 |
| PIM Shared Tree and Source Tree | 132 |
| Reverse Path Forwarding | 133 |
| RPF Check | 134 |
| Default PIM Routing Configuration | 135 |
| How to Configure PIM | 136 |
| Enabling PIM Stub Routing | 136 |
| Configuring a Rendezvous Point | 137 |
| Manually Assigning an RP to Multicast Groups | 138 |
| Setting Up Auto-RP in a New Internetwork | 140 |
| Adding Auto-RP to an Existing Sparse-Mode Cloud | 142 |
| Preventing Join Messages to False RPs | 145 |
| Filtering Incoming RP Announcement Messages | 145 |
| Configuring PIMv2 BSR | 147 |
| Defining the PIM Domain Border | 147 |
| Defining the IP Multicast Boundary | 148 |
| Configuring Candidate BSRs | 150 |
| Configuring the Candidate RPs | 151 |
| Configuring Sparse Mode with Auto-RP | 153 |
| Delaying the Use of PIM Shortest-Path Tree | 157 |
| Modifying the PIM Router-Query Message Interval | 159 |
| Verifying PIM Operations | 160 |
| Verifying IP Multicast Operation in a PIM-SM or a PIM-SSM Network | 160 |
| Verifying IP Multicast on the First Hop Router | 160 |
| Verifying IP Multicast on Routers Along the SPT | 162 |
| Verifying IP Multicast Operation on the Last Hop Router | 163 |
| Using PIM-Enabled Routers to Test IP Multicast Reachability | 167 |

- Configuring Routers to Respond to Multicast Pings 167
- Pinging Routers Configured to Respond to Multicast Pings 168
- Monitoring and Troubleshooting PIM 169
 - Monitoring PIM Information 169
 - Monitoring the RP Mapping and BSR Information 170
 - Troubleshooting PIMv1 and PIMv2 Interoperability Problems 170
- Configuration Examples for PIM 171
 - Example: Enabling PIM Stub Routing 171
 - Example: Verifying PIM Stub Routing 171
 - Example: Manually Assigning an RP to Multicast Groups 171
 - Example: Configuring Auto-RP 172
 - Example: Sparse Mode with Auto-RP 172
 - Example: Defining the IP Multicast Boundary to Deny Auto-RP Information 172
 - Example: Filtering Incoming RP Announcement Messages 172
 - Example: Preventing Join Messages to False RPs 173
 - Example: Configuring Candidate BSRs 173
 - Example: Configuring Candidate RPs 173
- Feature History for PIM 173

CHAPTER 8

- Configuring PIM MIB Extension for IP Multicast 175**
 - Information About PIM MIB Extension for IP Multicast 175
 - PIM MIB Extensions for SNMP Traps for IP Multicast 175
 - Benefits of PIM MIB Extensions 176
 - How to Configure PIM MIB Extension for IP Multicast 176
 - Enabling PIM MIB Extensions for IP Multicast 176
 - Configuration Examples for PIM MIB Extensions 177
 - Example Enabling PIM MIB Extensions for IP Multicast 177
 - Additional References for PIM MIB Extension for IP Multicast 178
 - Feature History for PIM MIB Extension for IP Multicast 178

CHAPTER 9

- Configuring SSM 181**
 - Prerequisites for Configuring SSM 181
 - Restrictions for Configuring SSM 181
 - Information About SSM 183

| | |
|--|-----|
| SSM Components Overview | 183 |
| SSM and Internet Standard Multicast (ISM) | 183 |
| SSM IP Address Range | 183 |
| SSM Operations | 184 |
| SSM Mapping | 184 |
| Static SSM Mapping | 184 |
| DNS-Based SSM Mapping | 185 |
| How to Configure SSM | 186 |
| Configuring SSM | 186 |
| Configuring Source Specific Multicast Mapping | 187 |
| Configuring Static SSM Mapping | 187 |
| Configuring DNS-Based SSM Mapping | 189 |
| Configuring Static Traffic Forwarding with SSM Mapping | 190 |
| Monitoring SSM | 192 |
| Monitoring SSM Mapping | 192 |
| Where to Go Next for SSM | 192 |
| Additional References for SSM | 193 |
| Feature History for SSM | 193 |

CHAPTER 10
Implementing IPv6 Multicast 195

| | |
|---|-----|
| Information About Implementing IPv6 Multicast Routing | 195 |
| IPv6 Multicast Overview | 195 |
| IPv6 Multicast Routing Implementation | 196 |
| IPv6 Multicast Listener Discovery Protocol | 196 |
| Multicast Queriers and Hosts | 196 |
| MLD Access Group | 197 |
| Explicit Tracking of Receivers | 197 |
| Protocol Independent Multicast | 197 |
| PIM-Sparse Mode | 197 |
| IPv6 BSR: Configure RP Mapping | 198 |
| PIM-Source Specific Multicast | 198 |
| Routable Address Hello Option | 199 |
| PIM IPv6 Stub Routing | 199 |
| Rendezvous Point | 200 |

| | |
|--|-----|
| Static Mroutes | 201 |
| MRIB | 201 |
| MFIB | 201 |
| MFIB | 201 |
| IPv6 Multicast VRF Lite | 202 |
| IPv6 Multicast Process Switching and Fast Switching | 202 |
| NSF and SSO Support In IPv6 Multicast | 203 |
| NTP in IPv6 | 203 |
| How to Implement IPv6 Multicast | 203 |
| Enabling IPv6 Multicast Routing | 203 |
| Customizing and Verifying the MLD Protocol | 204 |
| Customizing and Verifying MLD on an Interface | 204 |
| Implementing MLD Group Limits | 205 |
| Configuring Explicit Tracking of Receivers to Track Host Behavior | 207 |
| Resetting the MLD Traffic Counters | 207 |
| Clearing the MLD Interface Counters | 208 |
| Configuring PIM | 209 |
| Configuring PIM-SM and Displaying PIM-SM Information for a Group Range | 209 |
| Configuring PIM Options | 210 |
| Resetting the PIM Traffic Counters | 212 |
| Clearing the PIM Topology Table to Reset the MRIB Connection | 213 |
| Configuring PIM IPv6 Stub Routing | 214 |
| PIM IPv6 Stub Routing Configuration Guidelines | 214 |
| Default IPv6 PIM Routing Configuration | 215 |
| Enabling IPV6 PIM Stub Routing | 215 |
| Monitoring IPv6 PIM Stub Routing | 217 |
| Disabling Embedded RP Support in IPv6 PIM | 217 |
| Configuring a BSR | 218 |
| Configuring a BSR and Verifying BSR Information | 218 |
| Sending PIM RP Advertisements to the BSR | 219 |
| Configuring BSR for Use Within Scoped Zones | 220 |
| Configuring BSR Switches to Announce Scope-to-RP Mappings | 221 |
| Configuring SSM Mapping | 222 |
| Configuring Static Mroutes | 223 |

| | |
|--|-----|
| Using MFIB in IPv6 Multicast | 224 |
| Verifying MFIB Operation in IPv6 Multicast | 224 |
| Resetting MFIB Traffic Counters | 225 |
| Additional References | 226 |
| Feature History for IPv6 Multicast | 226 |

CHAPTER 11
Configuring MLD Snooping 227

| | |
|---|-----|
| Information About Configuring IPv6 MLD Snooping | 227 |
| Understanding MLD Snooping | 227 |
| MLD Messages | 228 |
| MLD Queries | 228 |
| Multicast Client Aging Robustness | 229 |
| Multicast Router Discovery | 229 |
| MLD Reports | 229 |
| MLD Done Messages and Immediate-Leave | 229 |
| Topology Change Notification Processing | 230 |
| How to Configure IPv6 MLD Snooping | 230 |
| Default MLD Snooping Configuration | 230 |
| MLD Snooping Configuration Guidelines | 231 |
| Enabling or Disabling MLD Snooping on the Switch | 231 |
| Enabling or Disabling MLD Snooping on a VLAN | 232 |
| Configuring a Static Multicast Group | 233 |
| Configuring a Multicast Router Port | 234 |
| Enabling MLD Immediate Leave | 235 |
| Configuring MLD Snooping Queries | 236 |
| Disabling MLD Listener Message Suppression | 238 |
| Monitoring MLD Snooping Information | 239 |
| Configuration Examples for Configuring MLD Snooping | 240 |
| Configuring a Static Multicast Group: Example | 240 |
| Configuring a Multicast Router Port: Example | 240 |
| Enabling MLD Immediate Leave: Example | 240 |
| Configuring MLD Snooping Queries: Example | 240 |
| Additional References | 241 |
| Feature History for MLD Snooping | 241 |

| | |
|-------------------|--|
| CHAPTER 12 | IP Multicast Optimization: Optimizing PIM Sparse Mode in a Large IP Multicast Deployment 243 |
| | Prerequisites for Optimizing PIM Sparse Mode in a Large IP Multicast Deployment 243 |
| | Information About Optimizing PIM Sparse Mode in a Large IP Multicast Deployment 243 |
| | PIM Registering Process 243 |
| | PIM Version 1 Compatibility 244 |
| | PIM Designated Router 244 |
| | PIM Sparse-Mode Register Messages 245 |
| | Preventing Use of Shortest-Path Tree to Reduce Memory Requirement 245 |
| | PIM Shared Tree and Source Tree - Shortest-Path Tree 245 |
| | Benefit of Preventing or Delaying the Use of the Shortest-Path Tree 246 |
| | How to Optimize PIM Sparse Mode in a Large IP Multicast Deployment 246 |
| | Optimizing PIM Sparse Mode in a Large Deployment 246 |
| | Configuration Examples for Optimizing PIM Sparse Mode in a Large Multicast Deployment 248 |
| | Optimizing PIM Sparse Mode in a Large IP Multicast Deployment Example 248 |
| | Additional References for IP Multicast Optimization: Optimizing PIM Sparse Mode in a Large IP Multicast Deployment 249 |
| | Feature History for IP Multicast Optimization: Optimizing PIM Sparse Mode in a Large IP Multicast Deployment 249 |

| | |
|-------------------|---|
| CHAPTER 13 | IP Multicast Optimization: Multicast Subsecond Convergence 251 |
| | Prerequisites for Multicast Subsecond Convergence 251 |
| | Restrictions for Multicast Subsecond Convergence 251 |
| | Information About Multicast Subsecond Convergence 251 |
| | Benefits of Multicast Subsecond Convergence 251 |
| | Multicast Subsecond Convergence Scalability Enhancements 252 |
| | PIM Router Query Messages 252 |
| | Reverse Path Forwarding 252 |
| | Topology Changes and Multicast Routing Recovery 252 |
| | How to Configure Multicast Subsecond Convergence 253 |
| | Modifying the PIM Router Query Message Interval 253 |
| | Verifying Multicast Subsecond Convergence Configurations 253 |
| | Configuration Examples for Multicast Subsecond Convergence 254 |
| | Modifying the PIM Router Query Message Interval Example 254 |

| | |
|---|---|
| Additional References for IP Multicast Optimization: Multicast Subsecond Convergence | 255 |
| Feature History for IP Multicast Optimization Multicast Subsecond Convergence | 255 |
| <hr/> | |
| CHAPTER 14 | IP Multicast Optimization: IP Multicast Load Splitting across Equal-Cost Paths 257 |
| Prerequisites for IP Multicast Load Splitting across Equal-Cost Paths | 257 |
| Information About IP Multicast Load Splitting across Equal-Cost Paths | 257 |
| Load Splitting Versus Load Balancing | 257 |
| Default Behavior for IP Multicast When Multiple Equal-Cost Paths Exist | 258 |
| Methods to Load Split IP Multicast Traffic | 259 |
| Overview of ECMP Multicast Load Splitting | 260 |
| ECMP Multicast Load Splitting Based on Source Address Using the S-Hash Algorithm | 260 |
| ECMP Multicast Load Splitting Based on Source and Group Address Using the Basic S-G-Hash Algorithm | 260 |
| Predictability As a By-Product of Using the S-Hash and Basic S-G-Hash Algorithms | 260 |
| Polarization As a By-Product of Using the S-Hash and Basic S-G-Hash Algorithms | 260 |
| ECMP Multicast Load Splitting Based on Source Group and Next-Hop Address | 261 |
| Effect of ECMP Multicast Load Splitting on PIM Neighbor Query and Hello Messages for RPF Path Selection | 262 |
| Effect of ECMP Multicast Load Splitting on the PIM Assert Process in PIM-SM and PIM-SSM | 263 |
| ECMP Multicast Load Splitting and Reconvergence When Unicast Routing Changes | 264 |
| Use of ECMP Multicast Load Splitting with Static Mroutes | 264 |
| Alternative Methods of Load Splitting IP Multicast Traffic | 264 |
| How to Load Split IP Multicast Traffic over ECMP | 265 |
| Enabling ECMP Multicast Load Splitting | 265 |
| Prerequisites for IP Multicast Load Splitting - ECMP | 265 |
| Restrictions for IP Multicast Load Splitting -ECMP | 266 |
| Enabling ECMP Multicast Load Splitting Based on Source Address | 266 |
| Enabling ECMP Multicast Load Splitting Based on Source and Group Address | 268 |
| Enabling ECMP Multicast Load Splitting Based on Source Group and Next-Hop Address | 269 |
| Configuration Examples for Load Splitting IP Multicast Traffic over ECMP | 271 |
| Example Enabling ECMP Multicast Load Splitting Based on Source Address | 271 |
| Example Enabling ECMP Multicast Load Splitting Based on Source and Group Address | 271 |
| Example Enabling ECMP Multicast Load Splitting Based on Source Group and Next-Hop Address | 271 |

| | |
|--|-----|
| Additional References for IP Multicast Optimization: IP Multicast Load Splitting across Equal-Cost Paths | 272 |
| Feature History for IP Multicast Optimization: IP Multicast Load Splitting across Equal-Cost Paths | 272 |

CHAPTER 15**IP Multicast Optimization: SSM Channel Based Filtering for Multicast 275**

| | |
|--|-----|
| Prerequisites for SSM Channel Based Filtering for Multicast Boundaries | 275 |
| Information About the SSM Channel Based Filtering for Multicast Boundaries | 275 |
| Rules for Multicast Boundaries | 275 |
| Benefits of SSM Channel Based Filtering for Multicast Boundaries | 276 |
| How to Configure SSM Channel Based Filtering for Multicast Boundaries | 276 |
| Configuring Multicast Boundaries | 276 |
| Configuration Examples for SSM Channel Based Filtering for Multicast Boundaries | 277 |
| Configuring the Multicast Boundaries Permitting and Denying Traffic Example | 277 |
| Configuring the Multicast Boundaries Permitting Traffic Example | 278 |
| Configuring the Multicast Boundaries Denying Traffic Example | 278 |
| Additional References for IP Multicast Optimization: SSM Channel-Based Filtering for Multicast | 278 |
| Feature History for IP Multicast Optimization: SSM Channel Based Filtering for Multicast | 279 |

CHAPTER 16**IP Multicast Optimization: IGMP State Limit 281**

| | |
|---|-----|
| Prerequisites for IGMP State Limit | 281 |
| Restrictions for IGMP State Limit | 281 |
| Information About IGMP State Limit | 281 |
| IGMP State Limit | 281 |
| IGMP State Limit Feature Design | 282 |
| Mechanics of IGMP State Limiters | 282 |
| How to Configure IGMP State Limit | 283 |
| Configuring IGMP State Limiters | 283 |
| Configuring Global IGMP State Limiters | 283 |
| Configuring Per Interface IGMP State Limiters | 283 |
| Configuration examples for IGMP State Limit | 284 |
| Configuring IGMP State Limiters Example | 285 |
| Additional References | 286 |
| Feature History for IP Multicast Optimization: IGMP State Limit | 286 |

| | | |
|-------------------|-------------------------------------|------------|
| CHAPTER 17 | Troubleshooting IP Multicast | 289 |
| | Overview | 289 |
| | Support Articles | 289 |
| | Feedback Request | 290 |
| | Disclaimer and Caution | 290 |



CHAPTER 1

IP Multicast Routing Technology Overview

- [Information About IP Multicast Technology, on page 1](#)
- [Additional References for IP Multicast, on page 16](#)
- [Feature History for IP Multicast Routing Technology, on page 19](#)

Information About IP Multicast Technology

This section provides information about IP multicast technology.

About IP Multicast

At one end of the IP communication spectrum is IP unicast, where a source IP host sends packets to a specific destination IP host. In IP unicast, the destination address in the IP packet is the address of a single, unique host in the IP network. These IP packets are forwarded across the network from the source to the destination host by devices. At each point on the path between source and destination, a device uses a unicast routing table to make unicast forwarding decisions, based on the IP destination address in the packet.

At the other end of the IP communication spectrum is an IP broadcast, where a source host sends packets to all hosts on a network segment. The destination address of an IP broadcast packet has the host portion of the destination IP address set to all ones and the network portion set to the address of the subnet. IP hosts, including devices, understand that packets, which contain an IP broadcast address as the destination address, are addressed to all IP hosts on the subnet. Unless specifically configured otherwise, devices do not forward IP broadcast packets, so IP broadcast communication is normally limited to a local subnet.

IP multicasting falls between IP unicast and IP broadcast communication. IP multicast communication enables a host to send IP packets to a group of hosts anywhere within the IP network. To send information to a specific group, IP multicast communication uses a special form of IP destination address called an IP multicast group address. The IP multicast group address is specified in the IP destination address field of the packet.

To multicast IP information, Layer 3 switches and devices must forward an incoming IP packet to all output interfaces that lead to members of the IP multicast group.

We tend to think of IP multicasting and video conferencing as the same thing. Although the first application in a network to use IP multicast is often video conferencing, video is only one of many IP multicast applications that can add value to a company's business model. Other IP multicast applications that have potential for improving productivity include multimedia conferencing, data replication, real-time data multicasts, and simulation applications.

Role of IP Multicast in Information Delivery

IP multicast is a bandwidth-conserving technology that reduces traffic by delivering a single stream of information simultaneously to potentially thousands of businesses and homes. Applications that take advantage of multicast include video conferencing, corporate communications, distance learning, and distribution of software, stock quotes, and news.

IP multicast routing enables a host (source) to send packets to a group of hosts (receivers) anywhere within the IP network by using a special form of IP address called the IP multicast group address. The sending host inserts the multicast group address into the IP destination address field of the packet and IP multicast routers and multilayer switches forward incoming IP multicast packets out all interfaces that lead to the members of the multicast group. Any host, regardless of whether it is a member of a group, can send to a group. However, only the members of a group receive the message. Controlling the transmission rate to a multicast group is not supported.

Multicast Group Transmission Scheme

IP communication consists of hosts that act as senders and receivers of traffic as shown in the first figure. Senders are called sources. Traditional IP communication is accomplished by a single host source sending packets to another single host (unicast transmission) or to all hosts (broadcast transmission). IP multicast provides a third scheme, allowing a host to send packets to a subset of all hosts (multicast transmission). This subset of receiving hosts is called a multicast group. The hosts that belong to a multicast group are called group members.

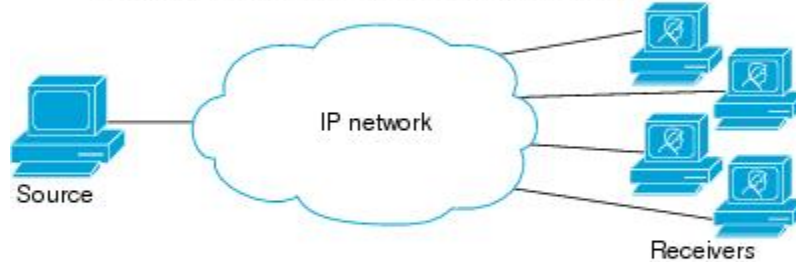
Multicast is based on this group concept. A multicast group is an arbitrary number of receivers that join a group in order to receive a particular data stream. This multicast group has no physical or geographical boundaries--the hosts can be located anywhere on the Internet or on any private internetwork. Hosts that are interested in receiving data from a source to a particular group must join that group. Joining a group is accomplished by a host receiver by way of the Internet Group Management Protocol (IGMP).

In a multicast environment, any host, regardless of whether it is a member of a group, can send to a group. However, only the members of a group can receive packets sent to that group. Multicast packets are delivered to a group using best-effort reliability, just like IP unicast packets.

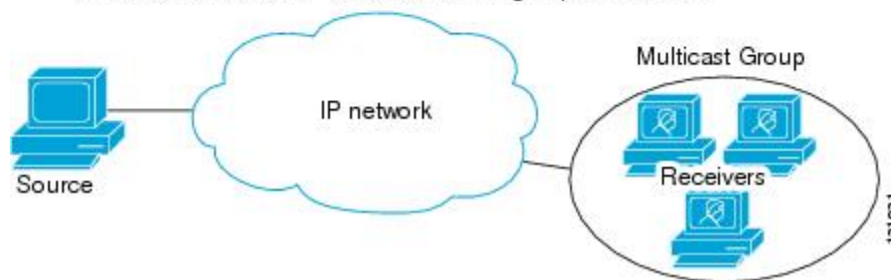
Unicast transmission—One host sends and the other receives.



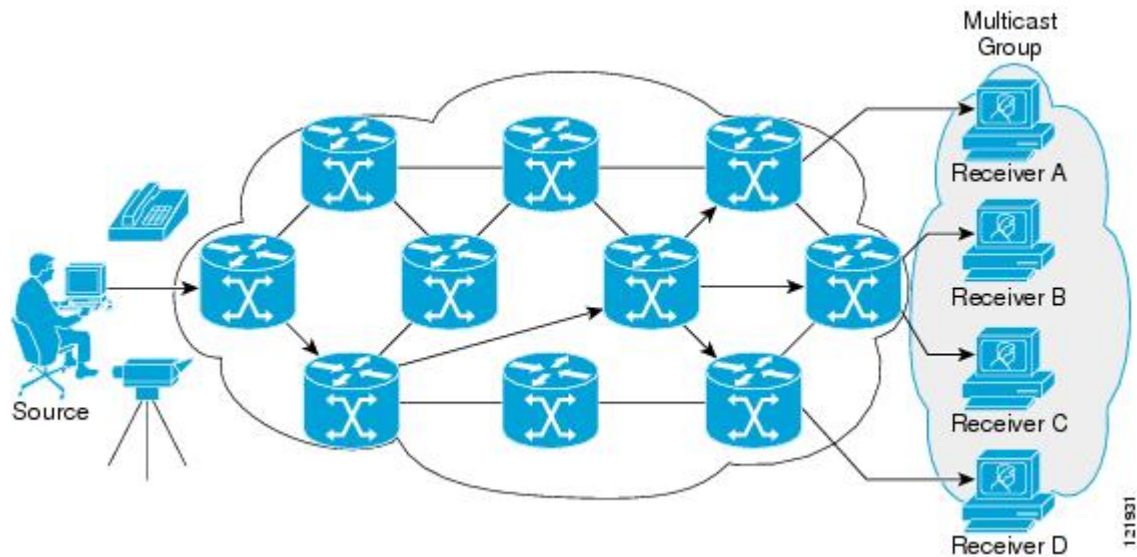
Broadcast transmission—One sender to all receivers.



Multicast transmission—One sender to a group of receivers.



In the next figure, the receivers (the designated multicast group) are interested in receiving the video data stream from the source. The receivers indicate their interest by sending an IGMP host report to the routers in the network. The routers are then responsible for delivering the data from the source to the receivers. The routers use Protocol Independent Multicast (PIM) to dynamically create a multicast distribution tree. The video data stream will then be delivered only to the network segments that are in the path between the source and the receivers.



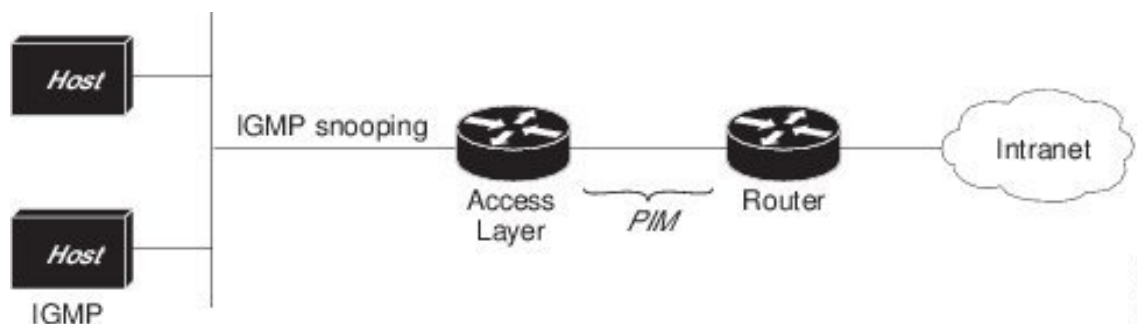
IP Multicast Routing Protocols

The software supports the following protocols to implement IP multicast routing:

- IGMP is used between hosts on a LAN and the routers (and multilayer devices) on that LAN to track the multicast groups of which hosts are members. To participate in IP multicasting, multicast hosts, routers, and multilayer devices must have the Internet Group Management Protocol (IGMP) operating.
- Protocol Independent Multicast (PIM) is used between routers so that they can track which multicast packets to forward to each other and to their directly connected LANs.
- IGMP Snooping is used for multicasting in a Layer 2 switching environment. It helps reduce the flooding of multicast traffic by dynamically configuring Layer 2 interfaces so that multicast traffic is forwarded to only those interfaces associated with IP multicast devices.

This figure shows where these protocols operate within the IP multicast environment.

Figure 1: IP Multicast Routing Protocols



According to IPv4 multicast standards, the MAC destination multicast address begins with 0100:5e and is appended by the last 23 bits of the IP address. For example, if the IP destination address is 239.1.1.39, the MAC destination address is 0100:5e01:0127.

A multicast packet is unmatched when the destination IPv4 address does not match the destination MAC address. The device forwards the unmatched packet in hardware based upon the MAC address table. If the destination MAC address is not in the MAC address table, the device floods the packet to the all port in the same VLAN as the receiving port.

Internet Group Management Protocol

IGMP messages are used by IP multicast hosts to send their local Layer 3 switch or router a request to join a specific multicast group and begin receiving multicast traffic. With some extensions in IGMPv2, IP hosts can also send a request to a Layer 3 switch or router to leave an IP multicast group and not receive the multicast group traffic.

Using the information obtained by using IGMP, a Layer 3 switch or router maintains a list of multicast group memberships on a per-interface basis. A multicast group membership is active on an interface if at least one host on the interface sends an IGMP request to receive multicast group traffic.

Protocol-Independent Multicast

Protocol-Independent Multicast (PIM) is protocol independent because it can leverage whichever unicast routing protocol is used to populate the unicast routing table, including EIGRP, OSPF, or static route, to support IP multicast.

PIM also uses a unicast routing table to perform the reverse path forwarding (RPF) check function instead of building a completely independent multicast routing table. PIM does not send and receive multicast routing updates between routers like other routing protocols do.

PIM Sparse Mode

PIM Sparse Mode (PIM-SM) uses a pull model to deliver multicast traffic. Only networks with active receivers that have explicitly requested the data are forwarded the traffic. PIM-SM is intended for networks with several different multicasts, such as desktop video conferencing and collaborative computing, that go to a small number of receivers and are typically in progress simultaneously.

Rendezvous Point

If you configure PIM to operate in sparse mode, you must also choose one or more devices to be rendezvous points (RPs). Senders to a multicast group use RPs to announce their presence. Receivers of multicast packets use RPs to learn about new senders. You can configure Cisco IOS software so that packets for a single multicast group can use one or more RPs.

The RP address is used by first hop devices to send PIM register messages on behalf of a host sending a packet to the group. The RP address is also used by last hop devices to send PIM join and prune messages to the RP to inform it about group membership. You must configure the RP address on all devices (including the RP device).

A PIM device can be an RP for more than one group. Only one RP address can be used at a time within a PIM domain for the same group. The conditions specified by the access list determine for which groups the device is an RP (as different groups can have different RPs).

IGMP Snooping

IGMP snooping is used for multicasting in a Layer 2 switching environment. With IGMP snooping, a Layer 3 switch or router examines Layer 3 information in the IGMP packets in transit between hosts and a device. When the switch receives the IGMP Host Report from a host for a particular multicast group, the switch adds

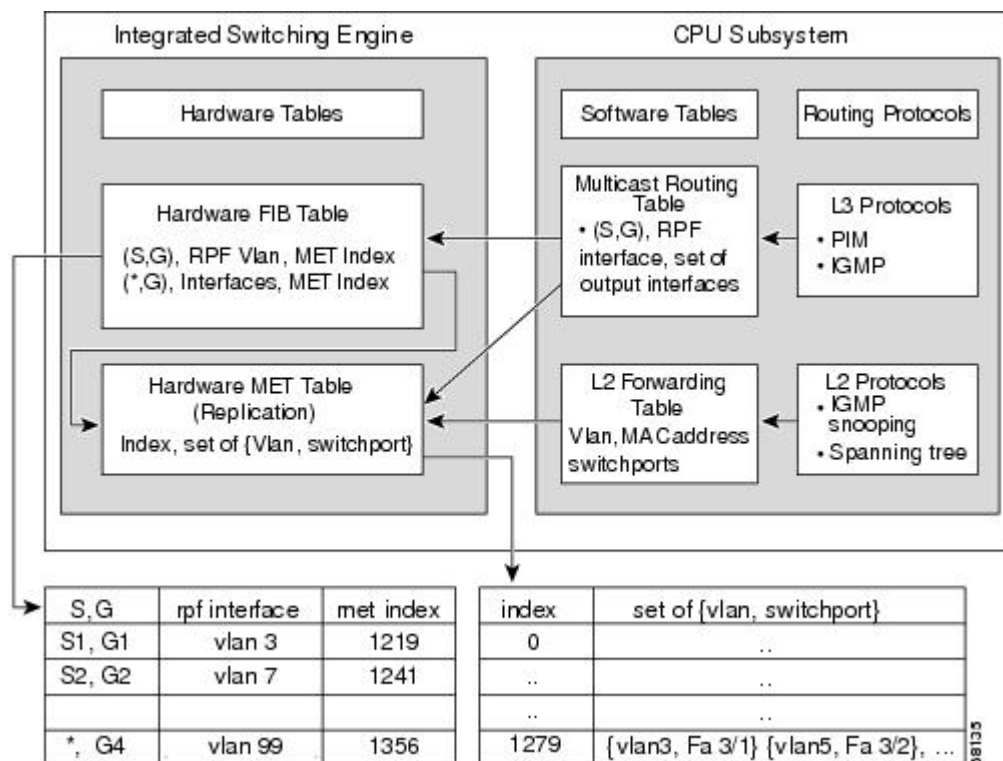
the host's port number to the associated multicast table entry. When the switch receives the IGMP Leave Group message from a host, it removes the host's port from the table entry.

Because IGMP control messages are transmitted as multicast packets, they are indistinguishable from multicast data if only the Layer 2 header is examined. A switch running IGMP snooping examines every multicast data packet to determine whether it contains any pertinent IGMP control information. If IGMP snooping is implemented on a low end switch with a slow CPU, performance could be severely impacted when data is transmitted at high rates.

IP Multicast Tables

The following illustration shows some key data structures that the device uses to forward IP multicast packets in hardware.

Figure 2: IP Multicast Tables and Protocols



The Integrated Switching Engine maintains the hardware FIB table to identify individual IP multicast routes. Each entry consists of a destination group IP address and an optional source IP address. Multicast traffic flows on primarily two types of routes: (S,G) and (*,G). The (S,G) routes flow from a source to a group based on the IP address of the multicast source and the IP address of the multicast group destination. Traffic on a (*,G) route flows from the PIM RP to all receivers of group G. Only sparse-mode groups use (*,G) routes. The Integrated Switching Engine hardware contains space for a total of 128,000 routes, which are shared by unicast routes, multicast routes, and multicast fast-drop entries.

Output interface lists are stored in the multicast expansion table (MET). The MET has room for up to 32,000 output interface lists. (For RET, we can have up to 102 K entries (32 K used for floodsets, 70,000 used for multicast entries)). The MET resources are shared by both Layer 3 multicast routes and by Layer 2 multicast entries. The actual number of output interface lists available in hardware depends on the specific configuration.

If the total number of multicast routes exceed 32,000, multicast packets might not be switched by the Integrated Switching Engine. They would be forwarded by the CPU subsystem at much slower speeds.



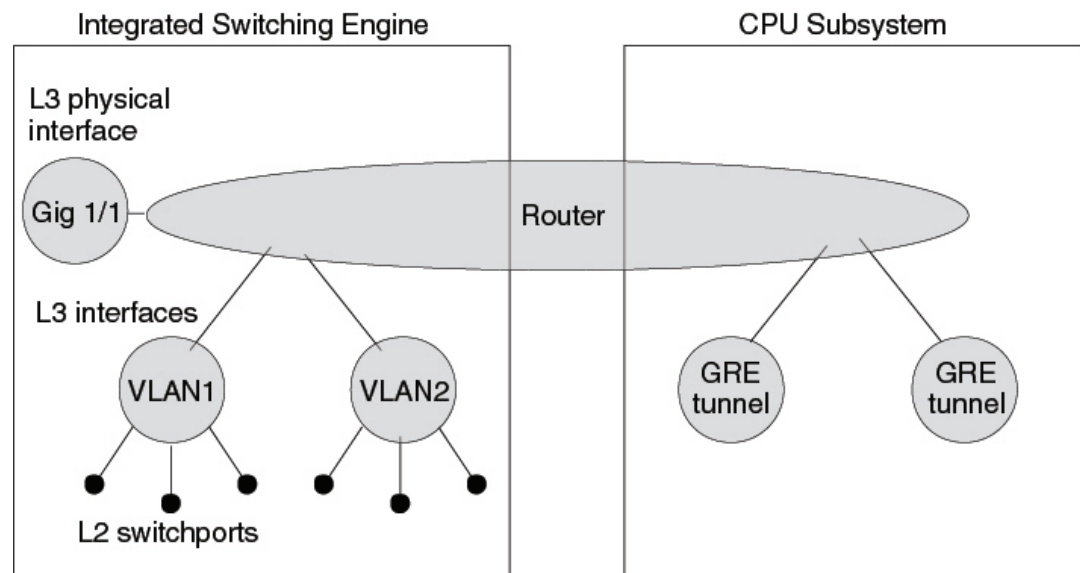
Note For RET, a maximum of 102 K entries is supported (32 K used for floodsets, 70 K used for multicast entries).

Hardware and Software Forwarding

The Integrated Switching Engine forwards the majority of packets in hardware at very high rates of speed. The CPU subsystem forwards exception packets in software. Statistical reports should show that the Integrated Switching Engine is forwarding the vast majority of packets in hardware.

The following illustration shows a logical view of the hardware and software forwarding components.

Figure 3: Hardware and Software Forwarding Components



68127

In the normal mode of operation, the Integrated Switching Engine performs inter-VLAN routing in hardware. The CPU subsystem supports generic routing encapsulation (GRE) tunnels for forwarding in software.

Replication is a particular type of forwarding where, instead of sending out one copy of the packet, the packet is replicated and multiple copies of the packet are sent out. At Layer 3, replication occurs only for multicast packets; unicast packets are never replicated to multiple Layer 3 interfaces. In IP multicasting, for each incoming IP multicast packet that is received, many replicas of the packet are sent out.

IP multicast packets can be transmitted on the following types of routes:

- Hardware routes
- Software routes
- Partial routes

Hardware routes occur when the Integrated Switching Engine hardware forwards all replicas of a packet. Software routes occur when the CPU subsystem software forwards all replicas of a packet. Partial routes occur

when the Integrated Switching Engine forwards some of the replicas in hardware and the CPU subsystem forwards some of the replicas in software.

Partial Routes



Note The conditions listed below cause the replicas to be forwarded by the CPU subsystem software, but the performance of the replicas that are forwarded in hardware is not affected.

The following conditions cause some replicas of a packet for a route to be forwarded by the CPU subsystem:

- The switch is configured with the **ip igmp join-group** command as a member of the IP multicast group on the RPF interface of the multicast source.
- The switch is the first-hop to the source in PIM sparse mode. The switch must send PIM-register messages to the RP.

Software Routes



Note If any one of the following conditions is configured on the RPF interface or the output interface, all replication of the output is performed in software.

The following conditions cause all replicas of a packet for a route to be forwarded by the CPU subsystem software:

- The interface is configured with multicast helper.
- The interface is a generic routing encapsulation (GRE) or Distance Vector Multicast Routing Protocol (DVMRP) tunnel.
- The interface uses non-Advanced Research Products Agency (ARPA) encapsulation.

The following packets are always forwarded in software:

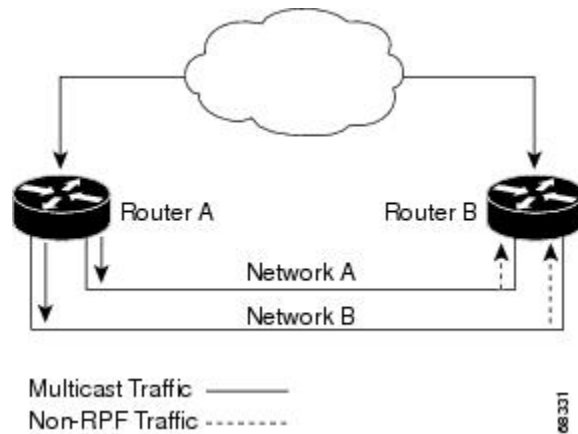
- Packets sent to multicast groups that fall into the range 224.0.0.* (where * is in the range from 0 to 255). This range is used by routing protocols. Layer 3 switching supports all other multicast group addresses.
- Packets with IP options.

Non-Reverse Path Forwarding Traffic

Traffic that fails an Reverse Path Forwarding (RPF) check is called non-RPF traffic. Non-RPF traffic is forwarded by the Integrated Switching Engine by filtering (persistently dropping) or rate limiting the non-RPF traffic.

In a redundant configuration where multiple Layer 3 switches or routers connect to the same LAN segment, only one device forwards the multicast traffic from the source to the receivers on the outgoing interfaces. The following illustration shows how non-RPF traffic can occur in a common network configuration.

Figure 4: Redundant Multicast Router Configuration in a Stub Network

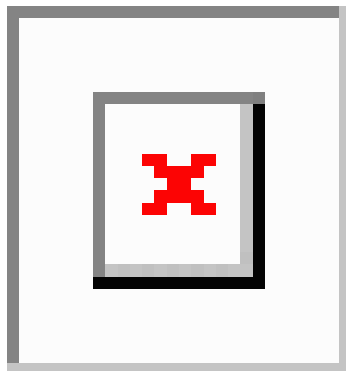


In this kind of topology, only Router A, the PIM designated router (PIM DR), forwards data to the common VLAN. Router B receives the forwarded multicast traffic, but must drop this traffic because it has arrived on the wrong interface and fails the RPF check. Traffic that fails the RPF check is called non-RPF traffic.

IP Multicast Boundary

As shown in the figure, address scoping defines domain boundaries so that domains with RPs that have the same IP address do not leak into each other. Scoping is performed on the subnet boundaries within large domains and on the boundaries between the domain and the Internet.

Figure 5: Address Scoping at Boundaries



You can set up an administratively scoped boundary on an interface for multicast group addresses using the **ip multicast boundary** command with the *access-list* argument. A standard access list defines the range of addresses affected. When a boundary is set up, no multicast data packets are allowed to flow across the boundary from either direction. The boundary allows the same multicast group address to be reused in different administrative domains.

The Internet Assigned Numbers Authority (IANA) has designated the multicast address range 239.0.0.0 to 239.255.255.255 as the administratively scoped addresses. This range of addresses can be reused in domains administered by different organizations. They would be considered local, not globally unique.

You can configure the **filter-autorp** keyword to examine and filter Auto-RP discovery and announcement messages at the administratively scoped boundary. Any Auto-RP group range announcements from the

Auto-RP packets that are denied by the boundary access control list (ACL) are removed. An Auto-RP group range announcement is permitted and passed by the boundary only if all addresses in the Auto-RP group range are permitted by the boundary ACL. If any address is not permitted, the entire group range is filtered and removed from the Auto-RP message before the Auto-RP message is forwarded.

IP Multicast Group Addressing

A multicast group is identified by its multicast group address. Multicast packets are delivered to that multicast group address. Unlike unicast addresses that uniquely identify a single host, multicast IP addresses do not identify a particular host. To receive the data sent to a multicast address, a host must join the group that address identifies. The data is sent to the multicast address and received by all the hosts that have joined the group indicating that they wish to receive traffic sent to that group. The multicast group address is assigned to a group at the source. Network administrators who assign multicast group addresses must make sure the addresses conform to the multicast address range assignments reserved by the Internet Assigned Numbers Authority (IANA).

IP Class D Addresses

IP multicast addresses have been assigned to the IPv4 Class D address space by IANA. The high-order four bits of a Class D address are 1110. Therefore, host group addresses can be in the range 224.0.0.0 to 239.255.255.255. A multicast address is chosen at the source (sender) for the receivers in a multicast group.



Note The Class D address range is used only for the group address or destination address of IP multicast traffic. The source address for multicast datagrams is always the unicast source address.

IP Multicast Address Scoping

The multicast address range is subdivided to provide predictable behavior for various address ranges and for address reuse within smaller domains. The table provides a summary of the multicast address ranges. A brief summary description of each range follows.

Table 1: Multicast Address Range Assignments

| Name | Range | Description |
|-------------------------------|------------------------------|--|
| Reserved Link-Local Addresses | 224.0.0.0 to 224.0.0.255 | Reserved for use by network protocols on a local network segment. |
| Globally Scoped Addresses | 224.0.1.0 to 238.255.255.255 | Reserved to send multicast data between organizations and across the Internet. |
| Source Specific Multicast | 232.0.0.0 to 232.255.255.255 | Reserved for use with the SSM datagram delivery model where data is forwarded only to receivers that have explicitly joined the group. |
| GLOP Addresses | 233.0.0.0 to 233.255.255.255 | Reserved for statically defined addresses by organizations that already have an assigned autonomous system (AS) domain number. |
| Limited Scope Address | 239.0.0.0 to 239.255.255.255 | Reserved as administratively or limited scope addresses for use in private multicast domains. |

Reserved Link-Local Addresses

The IANA has reserved the range 224.0.0.0 to 224.0.0.255 for use by network protocols on a local network segment. Packets with an address in this range are local in scope and are not forwarded by IP routers. Packets with link local destination addresses are typically sent with a time-to-live (TTL) value of 1 and are not forwarded by a router.

Within this range, reserved link-local addresses provide network protocol functions for which they are reserved. Network protocols use these addresses for automatic router discovery and to communicate important routing information. For example, Open Shortest Path First (OSPF) uses the IP addresses 224.0.0.5 and 224.0.0.6 to exchange link-state information.

IANA assigns single multicast address requests for network protocols or network applications out of the 224.0.1.xxx address range. Multicast routers forward these multicast addresses.



Note All the packets with reserved link-local addresses are punted to CPU by default in the ASR 903 RSP2 Module.

Globally Scoped Addresses

Addresses in the range 224.0.1.0 to 238.255.255.255 are called globally scoped addresses. These addresses are used to send multicast data between organizations across the Internet. Some of these addresses have been reserved by IANA for use by multicast applications. For example, the IP address 224.0.1.1 is reserved for Network Time Protocol (NTP).

Source Specific Multicast Addresses

Addresses in the range 232.0.0.0/8 are reserved for Source Specific Multicast (SSM) by IANA. In Cisco IOS software, you can use the **ip pim ssm** command to configure SSM for arbitrary IP multicast addresses also. SSM is an extension of Protocol Independent Multicast (PIM) that allows for an efficient data delivery mechanism in one-to-many communications. SSM is described in the [IP Multicast Delivery Modes, on page 14](#) section.

GLOP Addresses

GLOP addressing (as proposed by RFC 2770, GLOP Addressing in 233/8) proposes that the 233.0.0.0/8 range be reserved for statically defined addresses by organizations that already have an AS number reserved. This practice is called GLOP addressing. The AS number of the domain is embedded into the second and third octets of the 233.0.0.0/8 address range. For example, AS 62010 is written in hexadecimal format as F23A. Separating the two octets F2 and 3A results in 242 and 58 in decimal format. These values result in a subnet of 233.242.58.0/24 that would be globally reserved for AS 62010 to use.

Limited Scope Addresses

The range 239.0.0.0 to 239.255.255.255 is reserved as administratively or limited scoped addresses for use in private multicast domains. These addresses are constrained to a local group or organization. Companies, universities, and other organizations can use limited scope addresses to have local multicast applications that will not be forwarded outside their domain. Routers typically are configured with filters to prevent multicast traffic in this address range from flowing outside an autonomous system (AS) or any user-defined domain. Within an AS or domain, the limited scope address range can be further subdivided so that local multicast boundaries can be defined.



Note Network administrators may use multicast addresses in this range, inside a domain, without conflicting with others elsewhere in the Internet.

Layer 2 Multicast Addresses

Historically, network interface cards (NICs) on a LAN segment could receive only packets destined for their burned-in MAC address or the broadcast MAC address. In IP multicast, several hosts need to be able to receive a single data stream with a common destination MAC address. Some means had to be devised so that multiple hosts could receive the same packet and still be able to differentiate between several multicast groups. One method to accomplish this is to map IP multicast Class D addresses directly to a MAC address. Using this method, NICs can receive packets destined to many different MAC address.

Cisco Group Management Protocol (CGMP) is used on routers connected to Catalyst switches to perform tasks similar to those performed by IGMP. CGMP is necessary for those Catalyst switches that cannot distinguish between IP multicast data packets and IGMP report messages, both of which are addressed to the same group address at the MAC level.

Cisco Express Forwarding, MFIB, and Layer 2 Forwarding

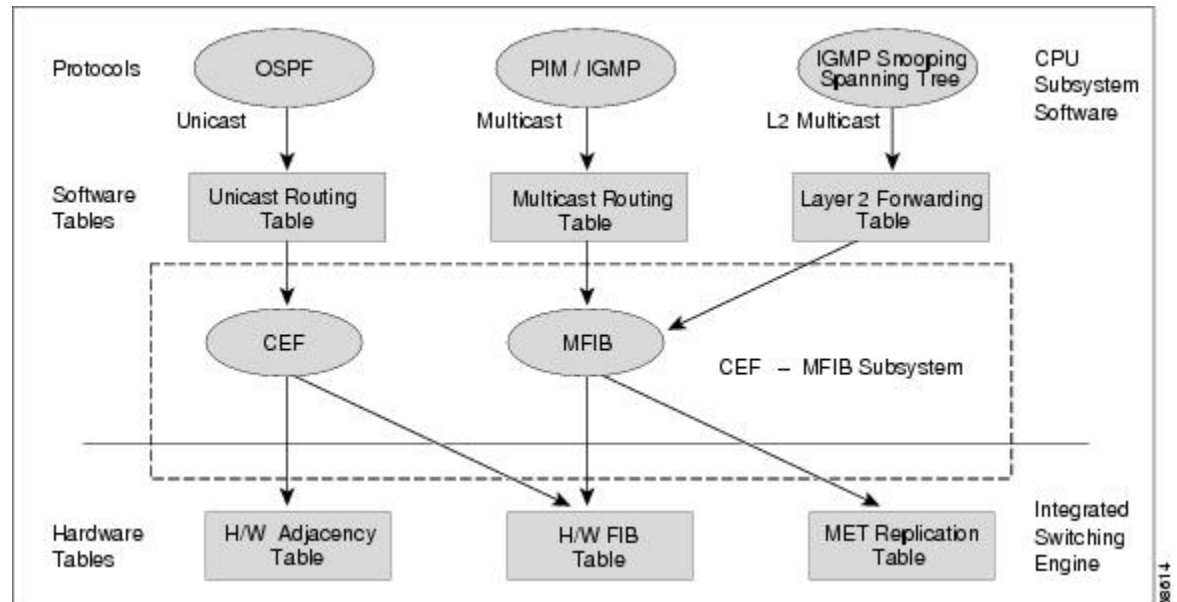
The implementation of IP multicast is an extension of centralized Cisco Express Forwarding. Cisco Express Forwarding extracts information from the unicast routing table, which is created by unicast routing protocols, such as OSPF, and EIGRP and loads it into the hardware

Forwarding Information Base (FIB). With the unicast routes in the FIB, when a route is changed in the upper-layer routing table, only one route needs to be changed in the hardware routing state. To forward unicast packets in hardware, the Integrated Switching Engine looks up source and destination routes in ternary content addressable memory (TCAM), takes the adjacency index from the hardware FIB, and gets the Layer 2 rewrite information and next-hop address from the hardware adjacency table.

The new Multicast Forwarding Information Base (MFIB) subsystem is the multicast analog of the unicast Cisco Express Forwarding. The MFIB subsystem extracts the multicast routes that PIM and IGMP create and refines them into a protocol-independent format for forwarding in hardware. The MFIB subsystem removes the protocol-specific information and leaves only the essential forwarding information. Each entry in the MFIB table consists of an (S,G) or (*,G) route, an input RPF VLAN, and a list of Layer 3 output interfaces. The MFIB subsystem, together with platform-dependent management software, loads this multicast routing information into the hardware FIB and Replica Expansion Table (RET). The device performs Layer 3 routing and Layer 2 bridging at the same time. There can be multiple Layer 2 switch ports on any VLAN interface.

The following illustration shows a functional overview of how a Cisco device combines unicast routing, multicast routing, and Layer 2 bridging information to forward in hardware:

Figure 6: Combining Cisco Express Forwarding, MFIB, and Layer 2 Forwarding Information in Hardware



Like the Cisco Express Forwarding unicast routes, the MFIB routes are Layer 3 and must be merged with the appropriate Layer 2 information. The following example shows an MFIB route:

```
(* ,203.0.113.1)
RPF interface is Vlan3
Output Interfaces are:
Vlan 1
Vlan 2
```

The route (*,203.0.113.1) is loaded in the hardware FIB table and the list of output interfaces is loaded into the MET. A pointer to the list of output interfaces, the MET index, and the RPF interface are also loaded in the hardware FIB with the (*,203.0.113.1) route. With this information loaded in hardware, merging of the Layer 2 information can begin. For the output interfaces on VLAN1, the Integrated Switching Engine must send the packet to all switch ports in VLAN1 that are in the spanning tree forwarding state. The same process applies to VLAN 2. To determine the set of switch ports in VLAN 2, the Layer 2 Forwarding Table is used.

When the hardware routes a packet, in addition to sending it to all of the switch ports on all output interfaces, the hardware also sends the packet to all switch ports (other than the one it arrived on) in the input VLAN. For example, assume that VLAN 3 has two switch ports in it, GigabitEthernet 3/1 and GigabitEthernet 3/2. If a host on GigabitEthernet 3/1 sends a multicast packet, the host on GigabitEthernet 3/2 might also need to receive the packet. To send a multicast packet to the host on GigabitEthernet 3/2, all of the switch ports in the ingress VLAN must be added to the port set that is loaded in the MET.

If VLAN 1 contains 1/1 and 1/2, VLAN 2 contains 2/1 and 2/2, and VLAN 3 contains 3/1 and 3/2, the MET chain for this route would contain these switch ports: (1/1,1/2,2/1,2/2,3/1, and 3/2).

If IGMP snooping is on, the packet should not be forwarded to all output switch ports on VLAN 2. The packet should be forwarded only to switch ports where IGMP snooping has determined that there is either a group member or router. For example, if VLAN 1 had IGMP snooping enabled, and IGMP snooping determined that only port 1/2 had a group member on it, then the MET chain would contain these switch ports: (1/1,1/2, 2/1, 2/2, 3/1, and 3/2).

IP Multicast Delivery Modes

IP multicast delivery modes differ only for the receiver hosts, not for the source hosts. A source host sends IP multicast packets with its own IP address as the IP source address of the packet and a group address as the IP destination address of the packet.

Source Specific Multicast

Source Specific Multicast (SSM) is a datagram delivery model that best supports one-to-many applications, also known as broadcast applications. SSM is a core network technology for the Cisco implementation of IP multicast targeted for audio and video broadcast application environments.

For the SSM delivery mode, an IP multicast receiver host must use IGMP Version 3 (IGMPv3) to subscribe to channel (S,G). By subscribing to this channel, the receiver host is indicating that it wants to receive IP multicast traffic sent by source host S to group G. The network will deliver IP multicast packets from source host S to group G to all hosts in the network that have subscribed to the channel (S, G).

SSM does not require group address allocation within the network, only within each source host. Different applications running on the same source host must use different SSM groups. Different applications running on different source hosts can arbitrarily reuse SSM group addresses without causing any excess traffic on the network.

Multicast Fast Drop

In IP multicast protocols, such as PIM-SM and PIM-DM, every (S,G) or (*,G) route has an incoming interface associated with it. This interface is referred to as the reverse path forwarding interface. In some cases, when a packet arrives on an interface other than the expected RPF interface, the packet must be forwarded to the CPU subsystem software to allow PIM to perform special protocol processing on the packet. One example of this special protocol processing that PIM performs is the PIM Assert protocol.

By default, the Integrated Switching Engine hardware sends all packets that arrive on a non-RPF interface to the CPU subsystem software. However, processing in software is not necessary in many cases, because these non-RPF packets are often not needed by the multicast routing protocols. The problem is that if no action is taken, the non-RPF packets that are sent to the software can overwhelm the CPU.

Instead of installing fast-drop entries, the Cisco device uses Dynamic Buffer Limiting (DBL). This flow-based congestion avoidance mechanism provides active queue management by tracking the queue length for each traffic flow. When the queue length of a flow exceeds its set limit, DBL drops packets. Rate DBL limits the non-rpf traffic to the cpu subsystem so that the CPU is not overwhelmed. The packets are rate limited per flow to the CPU. Because installing fast-drop entries in the CAM is inaccessibly, the number of fast-drop flows that can be handled by the switch need not be limited.

Protocol events, such as a link going down or a change in the unicast routing table, can impact the set of packets that can safely be fast dropped. A packet that was correctly fast dropped before might, after a topology change, need to be forwarded to the CPU subsystem software so that PIM can process it. The CPU subsystem software handles flushing fast-drop entries in response to protocol events so that the PIM code in IOS can process all the necessary RPF failures.

The use of fast-drop entries in the hardware is critical in some common topologies because you may have persistent RPF failures. Without the fast-drop entries, the CPU is exhausted by RPF failed packets that it did not need to process.

Multicast Forwarding Information Base

The Multicast Forwarding Information Base (MFIB) subsystem supports IP multicast routing in the Integrated Switching Engine hardware on Cisco devices. The MFIB logically resides between the IP multicast routing protocols in the CPU subsystem software (PIM, IGMP, MSDP, and DVMRP) and the platform-specific code that manages IP multicast routing in hardware. The MFIB translates the routing table information created by the multicast routing protocols into a simplified format that can be efficiently processed and used for forwarding by the Integrated Switching Engine hardware.

To display the information in the multicast routing table, use the **show ip mroute** command. To display the MFIB table information, use the **show ip mfib** command.

The MFIB table contains a set of IP multicast routes. IP multicast routes include (S,G) and (*,G). Each route in the MFIB table can have one or more optional flags associated with it. The route flags indicate how a packet that matches a route should be forwarded. For example, the Internal Copy (IC) flag on an MFIB route indicates that a process on the switch needs to receive a copy of the packet. The following flags can be associated with MFIB routes:

- Internal Copy (IC) flag—Sets on a route when a process on the router needs to receive a copy of all packets matching the specified route.
- Signalling (S) flag—Sets on a route when a process needs to be notified when a packet matching the route is received; the expected behavior is that the protocol code updates the MFIB state in response to receiving a packet on a signalling interface.
- Connected (C) flag—When set on an MFIB route, has the same meaning as the Signaling (S) flag, except that the C flag indicates that only packets sent by directly connected hosts to the route should be signaled to a protocol process.

A route can also have a set of optional flags associated with one or more interfaces. For example, an (S,G) route with the flags on VLAN 1 indicates how packets arriving on VLAN 1 should be handled, and whether packets matching the route should be forwarded onto VLAN 1. The per-interface flags supported in the MFIB include the following:

- Accepting (A)—Sets on the interface that is known in multicast routing as the RPF interface. A packet that arrives on an interface that is marked as Accepting (A) is forwarded to all Forwarding (F) interfaces.
- Forwarding (F)—Used in conjunction with the Accepting (A) flag as described above. The set of Forwarding interfaces that form what is often referred to as the multicast “olist” or output interface list.
- Signaling (S)—Sets on an interface when some multicast routing protocol process in Cisco IOS needs to be notified of packets arriving on that interface.



Note When PIM-SM routing is in use, the MFIB route might include an interface as in this example:

```
PimTunnel [1.2.3.4]
```

It is a virtual interface that the MFIB subsystem creates to indicate that packets are being tunneled to the specified destination address. A PimTunnel interface cannot be displayed with the normal **show interface** command.

S/M, 224/4

An (S/M, 224/4) entry is created in the MFIB for every multicast-enabled interface. This entry ensures that all packets sent by directly connected neighbors can be register-encapsulated to the PIM-SM RP. Typically, only a small number of packets are forwarded using the (S/M,224/4) route, until the (S,G) route is established by PIM-SM.

For example, on an interface with IP address 10.0.0.1 and netmask 255.0.0.0, a route is created matching all IP multicast packets in which the source address is anything in the class A network 10. This route can be written in conventional subnet/masklength notation as (10/8,224/4). If an interface has multiple assigned IP addresses, then one route is created for each such IP address.

Multicast High Availability

Cisco Catalyst 9200 Series Switches support multicast high availability, which ensures uninterrupted multicast traffic flow if a supervisor engine failure. MFIB states are synced to the standby supervisor engine before a switchover, ensuring NSF availability with a fast convergence upon switchover during a supervisor engine failure.

Multicast HA (SSO / NSF / ISSU) is supported for the PIM Sparse mode and SSM mode; and in Layer 2 for IGMP and MLD Snooping.

Additional References for IP Multicast

Related Documents

| Related Topic | Document Title |
|--|---|
| For complete syntax and usage information for the commands used in this chapter. | See the IP Multicast Routing Commands section of the <i>Command Reference (Catalyst 9200 Series Switches)</i> |

Standards and RFCs

| Standard/RFC | Title |
|--------------|---|
| RFC 1112 | <i>Host Extensions for IP Multicasting</i> |
| RFC 2236 | <i>Internet Group Management Protocol, Version 2</i> |
| RFC 4601 | <i>Protocol-Independent Multicast-Sparse Mode (PIM-SM): Protocol Specification</i> |
| RFC-9186 | Fast Failover in Protocol Independent Multicast - Sparse Mode (PIM-SM) Using Bidirectional Forwarding Detection (BFD) |
| RFC-9166 | A YANG Data Model for Internet Group Management Protocol (IGMP) and Multicast Listener Discovery (MLD) Snooping |
| RFC-8916 | A YANG Data Model for the Multicast Source Discovery Protocol (MSDP) |

| Standard/RFC | Title |
|--------------|---|
| RFC-8775 | PIM Designated Router Load Balancing |
| RFC-8736 | PIM Message Type Space Extension and Reserved Bits |
| RFC-8652 | A YANG Data Model for the Internet Group Management Protocol (IGMP) and Multicast Listener Discovery (MLD) |
| RFC-8364 | PIM Flooding Mechanism (PFM) and Source Discovery (SD) |
| RFC-8059 | PIM Join Attributes for Locator/ID Separation Protocol (LISP) Environments |
| RFC-7891 | Explicit Reverse Path Forwarding (RPF) Vector |
| RFC-7887 | Hierarchical Join/Prune Attributes |
| RFC-7761 | Protocol Independent Multicast - Sparse Mode (PIM-SM): Protocol Specification (Revised) |
| RFC-7063 | Survey Report on Protocol Independent Multicast - Sparse Mode (PIM-SM) Implementations and Deployments |
| RFC-6807 | Population Count Extensions to Protocol Independent Multicast (PIM) |
| RFC-6754 | Protocol Independent Multicast Equal-Cost Multipath (ECMP) Redirect |
| RFC-6559 | A Reliable Transport Mechanism for PIM |
| RFC-6420 | PIM Multi-Topology ID (MT-ID) Join Attribute |
| RFC-6395 | An Interface Identifier (ID) Hello Option for PIM |
| RFC-6226 | PIM Group-to-Rendezvous-Point Mapping |
| RFC-6166 | A Registry for PIM Message Types |
| RFC-5796 | Authentication and Confidentiality in Protocol Independent Multicast Sparse Mode (PIM-SM) Link-Local Messages |
| RFC-5496 | The Reverse Path Forwarding (RPF) Vector TLV Errata |
| RFC-5384 | The Protocol Independent Multicast (PIM) Join Attribute Format |
| RFC-5294 | Host Threats to Protocol Independent Multicast (PIM) |
| RFC-5240 | Protocol Independent Multicast (PIM) Bootstrap Router MIB Errata |
| RFC-5132 | IP Multicast MIB |
| RFC-5060 | Protocol Independent Multicast MIB Errata |

| Standard/RFC | Title |
|--------------|---|
| RFC-5059 | Bootstrap Router (BSR) Mechanism for Protocol Independent Multicast (PIM) |
| RFC-5015 | Bidirectional Protocol Independent Multicast (BIDIR-PIM) |
| RFC-4610 | Anycast-RP Using Protocol Independent Multicast (PIM) |
| RFC-4601 | Protocol Independent Multicast - Sparse Mode (PIM-SM): Protocol Specification |
| RFC-4541 | Considerations for Internet Group Management Protocol (IGMP) and Multicast Listener Discovery (MLD) Snooping Switches |
| RFC-4291 | IP Version 6 Addressing Architecture |
| RFC-3973 | Protocol Independent Multicast - Dense Mode (PIM-DM): Protocol Specification |
| RFC-3618 | Multicast Source Discovery Protocol (MSDP) |
| RFC-3569 | An Overview of Source-Specific Multicast (SSM) |
| RFC-3446 | Anycast Rendezvous Point (RP) mechanism using Protocol Independent Multicast (PIM) and Multicast Source Discovery Protocol (MSDP) |
| RFC-3376 | Internet Group Management Protocol - Version 3 |
| RFC-3048 | Reliable Multicast Transport Building Blocks for One-to-Many Bulk-Data Transfer |
| RFC-2858 | Multiprotocol Extensions for BGP-4 |
| RFC-2715 | Interoperability Rules for Multicast Routing Protocols |
| RFC-2710 | Multicast Listener Discovery (MLD) for IPv6 |
| RFC-2674 | Definitions of Managed Objects for Bridges with Traffic Classes - Multicast Filtering and Virtual LAN Extensions |
| RFC-2627 | Key Management for Multicast: Issues and Architectures |
| RFC-2432 | Terminology for IP Multicast Benchmarking |
| RFC-2375 | IPv6 Multicast Address Assignments |
| RFC-2365 | Administratively Scoped IP Multicast |
| RFC-2362 | Protocol Independent Multicast-Sparse Mode (PIM-SM): Protocol Specification |
| RFC-2357 | IETF Criteria for Evaluating Reliable Multicast Transport and Application Protocols |

| Standard/RFC | Title |
|--------------|---|
| RFC-2236 | Internet Group Management Protocol - Version 2 |
| RFC-2201 | Core Based Trees (CBT) Multicast Routing Architecture |
| RFC-2189 | Core Based Trees (CBT version 2) Multicast Routing |
| RFC-1949 | Scalable Multicast Key Distribution |
| RFC-1584 | Multicast Extensions to OSPF |
| RFC-1458 | Requirements for Multicast Protocols |

Feature History for IP Multicast Routing Technology

This table provides release and related information for features explained in this module.

These features are available on all releases subsequent to the one they were introduced in, unless noted otherwise.

| Release | Feature | Feature Information |
|--------------------------|---------------------------------|--|
| Cisco IOS XE Fuji 16.9.2 | IP Multicast Routing Technology | IP multicast is a bandwidth-conserving technology that reduces traffic by delivering a single stream of information simultaneously to potentially thousands of businesses and homes. |

Use Cisco Feature Navigator to find information about platform and software image support. To access Cisco Feature Navigator, go to [Cisco Feature Navigator](#).



CHAPTER 2

Configuring Basic IP Multicast Routing

- [Information About Basic IP Multicast Routing, on page 21](#)
- [How to Configure Basic IP Multicast Routing, on page 22](#)
- [Monitoring and Maintaining Basic IP Multicast Routing, on page 32](#)
- [Configuration Examples for Basic IP Multicast Routing, on page 34](#)
- [Additional References for Basic IP Multicast Routing, on page 35](#)
- [Feature History for Basic IP Multicast Routing, on page 35](#)

Information About Basic IP Multicast Routing

IP multicasting is an efficient way to use network resources, especially for bandwidth-intensive services such as audio and video. IP multicast routing enables a host (source) to send packets to a group of hosts (receivers) anywhere within the IP network by using a special form of IP address called the IP multicast group address.

The sending host inserts the multicast group address into the IP destination address field of the packet, and IP multicast routers and multilayer forward incoming IP multicast packets out all interfaces that lead to members of the multicast group. Any host, regardless of whether it is a member of a group, can send to a group. However, only the members of a group receive the message.

Multicast Forwarding Information Base Overview

The uses the Multicast Forwarding Information Base (MFIB) architecture and the Multicast Routing Information Base (MRIB) for IP multicast.

The MFIB architecture provides both modularity and separation between the multicast control plane (Protocol Independent Multicast [PIM] and Internet Group Management Protocol [IGMP]) and the multicast forwarding plane (MFIB). This architecture is used in Cisco IOS IPv6 multicast implementations.

MFIB itself is a multicast routing protocol independent forwarding engine; that is, it does not depend on PIM or any other multicast routing protocol. It is responsible for:

- Forwarding multicast packets
- Registering with the MRIB to learn the entry and interface flags set by the control plane
- Handling data-driven events that must be sent to the control plane
- Maintaining counts, rates, and bytes of received, dropped, and forwarded multicast packets

The MRIB is the communication channel between MRIB clients. Examples of MRIB clients are PIM, IGMP, the multicast routing (mroute) table, and the MFIB.

Default IP Multicast Routing Configuration

This table displays the default IP multicast routing configuration.

Table 2: Default IP Multicast Routing Configuration

| Feature | Default Setting |
|-----------------------------------|-----------------------------|
| Multicast routing | Disabled on all interfaces. |
| PIM version | Version 2. |
| PIM mode | No mode is defined. |
| PIM stub routing | None configured. |
| PIM RP address | None configured. |
| PIM domain border | Disabled. |
| PIM multicast boundary | None. |
| Candidate BSRs | Disabled. |
| Candidate RPs | Disabled. |
| Shortest-path tree threshold rate | 0 kb/s. |
| PIM router query message interval | 30 seconds. |

How to Configure Basic IP Multicast Routing

This section provides information about configuring basic IP multicast routing.

Configuring Basic IP Multicast Routing

Before you begin



Note By default, multicast routing is disabled, and there is no default mode setting. To enable multicast routing, use the **ip multicast-routing** command.

You must configure the PIM version and the PIM mode. The switch populates its multicast routing table and forwards multicast packets it receives from its directly connected LANs according to the mode setting.

In populating the multicast routing table, dense-mode interfaces are always added to the table. Sparse-mode interfaces are added to the table only when periodic join messages are received from downstream devices or

when there is a directly connected member on the interface. When forwarding from a LAN, sparse-mode operation occurs if there is an RP known for the group. If so, the packets are encapsulated and sent toward the RP. When no RP is known, the packet is flooded in a dense-mode fashion. The multicast source address must be on the directly connected incoming interface (that is part of the same subnet) of the first-hop router for both PIM dense mode and PIM any-source multicast mode. If the multicast traffic from a specific source is sufficient, the receiver's first-hop router might send join messages toward the source to build a source-based distribution tree.

Procedure

| | Command or Action | Purpose |
|---------------|---|--|
| Step 1 | enable Example: Device> enable | Enables privileged EXEC mode. Enter your password, if prompted. |
| Step 2 | configure terminal Example: Device# configure terminal | Enters global configuration mode. |
| Step 3 | interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet 1/0/1 | Specifies the Layer 3 interface on which you want to enable multicast routing, and enters interface configuration mode. The specified interface must be one of the following: <ul style="list-style-type: none"> • A routed port—A physical port that has been configured as a Layer 3 port by entering the no switchport interface configuration command. You will also need to enable IP PIM sparse-dense-mode on the interface, and join the interface as a statically connected member to an IGMP static group. • An SVI—A VLAN interface created by using the interface vlan <i>vlan-id</i> global configuration command. You will also need to enable IP PIM sparse-dense-mode on the VLAN, join the VLAN as a statically connected member to an IGMP static group, and then enable IGMP snooping on the VLAN, the IGMP static group, and physical interface. These interfaces must have IP addresses assigned to them. |

| | Command or Action | Purpose |
|---------------|--|--|
| Step 4 | <p>ip pim {dense-mode sparse-mode sparse-dense-mode}</p> <p>Example:</p> <pre>Device(config-if) # ip pim sparse-mode</pre> | <p>Enables a PIM mode on the interface.</p> <p>By default, no mode is configured.</p> <p>The keywords have these meanings:</p> <ul style="list-style-type: none"> • dense-mode—Enables dense mode of operation. • sparse-mode—Enables sparse mode of operation. If you configure sparse mode, you must also configure an RP. • sparse-dense-mode—Causes the interface to be treated in the mode in which the group belongs. Sparse-dense mode is the recommended setting. <p>Note To disable PIM on an interface, use the no ip pim interface configuration command.</p> |
| Step 5 | <p>end</p> <p>Example:</p> <pre>Device(config-if) # end</pre> | Returns to privileged EXEC mode. |
| Step 6 | <p>show running-config</p> <p>Example:</p> <pre>Device# show running-config</pre> | Verifies your entries. |
| Step 7 | <p>copy running-config startup-config</p> <p>Example:</p> <pre>Device# copy running-config startup-config</pre> | (Optional) Saves your entries in the configuration file. |

Configuring IP Multicast Forwarding

You can use the following procedure to configure IPv4 Multicast Forwarding Information Base (MFIB) interrupt-level IP multicast forwarding of incoming packets or outgoing packets on the device.



Note After you have enabled IP multicast routing by using the **ip multicast-routing** command, IPv4 multicast forwarding is enabled.

Procedure

| | Command or Action | Purpose |
|---------------|---|--|
| Step 1 | enable Example: Device> enable | Enables privileged EXEC mode. Enter your password, if prompted. |
| Step 2 | configure terminal Example: Device# configure terminal | Enters global configuration mode. |
| Step 3 | ip mfib Example: Device(config)# ip mfib | Enables IP multicast forwarding. |
| Step 4 | exit Example: Device(config)# exit | Returns to privileged EXEC mode. |
| Step 5 | show running-config Example: Device# show running-config | Verifies your entries. |
| Step 6 | copy running-config startup-config Example: Device# copy running-config startup-config | (Optional) Saves your entries in the configuration file. |

Configuring a Static Multicast Route (mroute)

- Static mroutes are used to calculate RPF information, not to forward traffic.
- Static mroutes cannot be redistributed.

Static mroutes are strictly local to the device on which they are defined. Because Protocol Independent Multicast (PIM) does not have its own routing protocol, there is no mechanism to distribute static mroutes throughout the network. Consequently, the administration of static mroutes tends to be more complicated than the administration of unicast static routes.

When static mroutes are configured, they are stored on the device in a separate table referred to as the static mroute table. When configured, the **ip mroute** command enters a static mroute into the static mroute table for the source address or source address range specified for the source-address and mask arguments. Sources that match the source address or that fall in the source address range specified for the source-address argument will RPF to either the interface associated with the IP address specified for the *rpf-address* argument or the local interface on the device specified for the *interface-type* and *interface-number* arguments. If an IP address is specified for the *rpf-address* argument, a recursive lookup is done from the unicast routing table on this address to find the directly connected neighbor.

If there are multiple static mroutes configured, the device performs a longest-match lookup of the mroute table. When the mroute with the longest match (of the source-address) is found, the search terminates and the information in the matching static mroute is used. The order in which the static mroutes are configured is not important.

The administrative distance of an mroute may be specified for the optional distance argument. If a value is not specified for the distance argument, the distance of the mroute defaults to zero. If the static mroute has the same distance as another RPF source, the static mroute will take precedence. There are only two exceptions to this rule: directly connected routes and the default unicast route.

Procedure

| | Command or Action | Purpose |
|---------------|---|---|
| Step 1 | enable Example: Device> enable | Enables privileged EXEC mode. Enter your password, if prompted. |
| Step 2 | configure terminal Example: Device# configure terminal | Enters global configuration mode. |
| Step 3 | ip mroute [vrf vrf-name] <i>source-address mask</i> { fallback-lookup { global vrf vrf-name } [<i>protocol</i>] { <i>rpf-address</i> <i>interface-type interface-number</i> } } [distance] Example: Device (config) # ip mroute 10.1.1.1 255.255.255.255 10.2.2.2 | The source IP address 10.1.1.1 is configured to be reachable through the interface associated with IP address 10.2.2.2. |
| Step 4 | exit Example: Device (config) # exit | Returns to privileged EXEC mode. |
| Step 5 | show running-config Example: Device# show running-config | (Optional) Verifies your entries. |
| Step 6 | copy running-config startup-config Example: Device# copy running-config startup-config | (Optional) Saves your entries in the configuration file. |

Configuring Multicast VRFs

Procedure

| | Command or Action | Purpose |
|---------------|--|--|
| Step 1 | configure terminal Example: Switch# configure terminal | Enters global configuration mode. |
| Step 2 | ip routing Example: Switch(config)# ip routing | Enables IP routing. |
| Step 3 | ip vrf vrf-name Example: Switch(config)# ip vrf vrf-name | Configures a VRF table and enters VRF configuration mode. |
| Step 4 | ip multicast-routing vrf vrf-name Example: Switch(config-vrf)# ip multicast-routing vrf vrf-name | (Optional) Enables global multicast routing for VRF table. |
| Step 5 | rd route-distinguisher Example: Switch (config-vrf)# rd route-distinguisher | Creates a VRF table by specifying a route distinguisher. Enter either an AS number and an arbitrary number (xxx:y) or an IP address and arbitrary number (A.B.C.D:y). |
| Step 6 | route-target {export import both} route-target-ext-community Example: Switch(config-vrf)# route-target {export import both} route-target-ext-community | Creates a list of import, export, or import and export route target communities for the specified VRF. Enter either an AS system number and an arbitrary number (xxx:y) or an IP address and an arbitrary number (A.B.C.D:y). The route-target-ext-community value should be the same as the route-distinguisher value entered in Step 4. |
| Step 7 | import map route-map Example: Switch(config-vrf)# import map route-map | (Optional) Associates a route map with the VRF. |
| Step 8 | interface interface-id Example: Switch (config)# interface interface-id | Enters interface configuration mode and specifies the Layer 3 interface to be associated with the VRF. The interface can be a routed port or a SVI. |

| | Command or Action | Purpose |
|----------------|---|--|
| Step 9 | vrf forwarding <i>vrf-name</i> Example: Switch (config-sg-tacacs+)# vrf forwarding vrf-name | Associates the VRF with the Layer 3 interface. |
| Step 10 | ip address <i>ip-address mask</i> Example: Switch (config-if)# ip address ip-address mask | Configures IP address for the Layer 3 interface. |
| Step 11 | ip pim sparse-mode Example: Switch(config-if)# ip pim sparse-mode | Enables PIM on the VRF-associated Layer 3 interface. |
| Step 12 | end Example: Switch(config-if)# end | Returns to privileged EXEC mode. |
| Step 13 | show ip vrf [brief detail interfaces] [vrf-name] Example: show ip vrf [brief detail interfaces] [vrf-name] | Verifies the configuration. Display information about the configured VRFs. |
| Step 14 | copy running-config startup-config Example: Switch# copy running-config startup-config | (Optional) Saves your entries in the configuration file. |

Example

The following example shows how to configure multicast within a VRF table:

```
Switch(config)# ip routing
Switch(config)# ip vrf multiVrfA
Switch(config-vrf)# ip multicast-routing vrf multiVrfA
Switch(config-vrf)# interface GigabitEthernet3/1/0
Switch(config-if)# vrf forwarding multiVrfA
Switch(config-if)# ip address 172.21.200.203 255.255.255.0
Switch(config-if)# ip pim sparse-mode
```

Configuring Optional IP Multicast Routing Features

This section provides information about configuring optional IP multicast routing features.

Defining the IP Multicast Boundary

You define a multicast boundary to prevent Auto-RP messages from entering the PIM domain. You create an access list to deny packets destined for 224.0.1.39 and 224.0.1.40, which carry Auto-RP information.

This procedure is optional.

Procedure

| | Command or Action | Purpose |
|---------------|---|--|
| Step 1 | enable Example: Device> enable | Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted. |
| Step 2 | configure terminal Example: Device# configure terminal | Enters global configuration mode. |
| Step 3 | access-list <i>access-list-number</i> deny <i>source</i> [<i>source-wildcard</i>] Example: Device(config)# access-list 12 deny 224.0.1.39 access-list 12 deny 224.0.1.40 | Creates a standard access list, repeating the command as many times as necessary. <ul style="list-style-type: none"> • For <i>access-list-number</i>, the range is 1 to 99. • The deny keyword denies access if the conditions are matched. • For <i>source</i>, enter multicast addresses 224.0.1.39 and 224.0.1.40, which carry Auto-RP information. • (Optional) For <i>source-wildcard</i>, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. The access list is always terminated by an implicit deny statement for everything. |
| Step 4 | interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet 1/0/1 | Specifies the interface to be configured, and enters interface configuration mode. |
| Step 5 | ip multicast boundary <i>access-list-number</i> Example: Device(config-if)# ip multicast boundary 12 | Configures the boundary, specifying the access list you created in Step 2. |
| Step 6 | end Example: | Returns to privileged EXEC mode. |

| | Command or Action | Purpose |
|---------------|---|--|
| | Device(config)# end | |
| Step 7 | show running-config Example: Device# show running-config | Verifies your entries. |
| Step 8 | copy running-config startup-config Example: Device# copy running-config startup-config | (Optional) Saves your entries in the configuration file. |

Configuring sdr Listener Support

This section provides information about configuring sdr listener support.

Enabling sdr Listener Support

By default, the device does not listen to session directory advertisements. This procedure is optional.

Procedure

| | Command or Action | Purpose |
|---------------|---|---|
| Step 1 | enable Example: Device> enable | Enables privileged EXEC mode. Enter your password, if prompted. |
| Step 2 | configure terminal Example: Device# configure terminal | Enters global configuration mode. |
| Step 3 | interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet 1/0/1 | Specifies the interface to be enabled for sdr, and enters interface configuration mode. |
| Step 4 | ip sap listen Example: Device(config-if)# ip sap listen | Enables the device software to listen to session directory announcements. |
| Step 5 | end Example: Device(config-if)# end | Returns to privileged EXEC mode. |
| Step 6 | show running-config Example: Device# show running-config | Verifies your entries. |

| | Command or Action | Purpose |
|---------------|---|--|
| Step 7 | copy running-config startup-config Example: Device# <code>copy running-config startup-config</code> | (Optional) Saves your entries in the configuration file. |

Limiting How Long an sdr Cache Entry Exists

By default, entries are never deleted from the sdr cache. You can limit how long the entry remains active so that if a source stops advertising SAP information, old advertisements are not unnecessarily kept.

This procedure is optional.

Procedure

| | Command or Action | Purpose |
|---------------|---|--|
| Step 1 | enable Example: Device> <code>enable</code> | Enables privileged EXEC mode. Enter your password, if prompted. |
| Step 2 | configure terminal Example: Device# <code>configure terminal</code> | Enters global configuration mode. |
| Step 3 | ip sap cache-timeout <i>minutes</i> Example: Device(config)# <code>ip sap cache-timeout 30</code> | Limits how long a Session Announcement Protocol (SAP) cache entry stays active in the cache. By default, entries are never deleted from the cache. For <i>minutes</i> , the range is 1 to 1440 minutes (24 hours). |
| Step 4 | end Example: Device(config)# <code>end</code> | Returns to privileged EXEC mode. |
| Step 5 | show running-config Example: Device# <code>show running-config</code> | Verifies your entries. |
| Step 6 | show ip sap Example: Device# <code>show ip sap</code> | Displays the SAP cache. |
| Step 7 | copy running-config startup-config Example: | (Optional) Saves your entries in the configuration file. |

| | Command or Action | Purpose |
|--|---|---------|
| | Device# <code>copy running-config startup-config</code> | |

Monitoring and Maintaining Basic IP Multicast Routing

Clearing Caches, Tables, and Databases

You can remove all contents of a particular cache, table, or database. Clearing a cache, table, or database might be necessary when the contents of the particular structure are or suspected to be invalid.

You can use any of the privileged EXEC commands in the following table to clear IP multicast caches, tables, and databases.

Table 3: Commands for Clearing Caches, Tables, and Databases

| Command | Purpose |
|---|--|
| <code>clear ip igmp group {group [hostname IP address] vrf name group [hostname IP address]}</code> | Deletes entries from the IGMP table. |
| <code>clear ip mroute { * [hostname IP address] vrf name group [hostname IP address]}</code> | Deletes entries from the IP multicast routing table. |
| <code>clear ip sap [group-address "session-name"]</code> | Deletes the Session Direct cache entry. |

Displaying System and Network Statistics

You can display specific statistics, such as the contents of IP routing tables, caches, and databases.



Note This release does not support per-route statistics.

You can display information to learn resource usage and solve network problems. You can also display information about node reachability and discover the routing path that packets of your device are taking through the network.

You can use any of the privileged EXEC commands in the following table to display various routing statistics.

Table 4: Commands for Displaying System and Network Statistics

| Command | Purpose |
|---|---|
| <code>ping [group-name group-address]</code> | Sends an ICMP Echo Request to a multicast group. |
| <code>show ip igmp filter</code> | Displays IGMP filter information. |
| <code>show ip igmp groups [group-name group-address type-number]</code> | Displays the multicast groups that are directly connected to the interface. |

| Command | Purpose |
|---|--|
| show ip igmp interface [<i>type number</i>] | Displays multicast-related information a |
| show ip igmp profile [<i>profile_number</i>] | Displays IGMP profile information. |
| show ip igmp ssm-mapping [<i>hostname/IP address</i>] | Displays IGMP SSM mapping informati |
| show ip igmp static-group { class-map [interface [<i>type</i>]] } | Displays static group information. |
| show ip igmp membership [<i>name/group address</i> all tracked] | Displays IGMP membership information |
| show ip igmp vrf | Displays the selected VPN Routing/Forv |
| show ip mfib [<i>type number</i>] | Displays the IP multicast forwarding inf |
| show ip mrib { client route vrf } | Displays the multicast routing informati |
| show ip mrm { interface manager status-report } | Displays the IP multicast routing monito |
| show ip mroute [<i>group-name</i> <i>group-address</i>] [<i>source</i>] [count interface proxy pruned summary verbose] | Displays the contents of the IP multicas |
| show ip msdp { count peer rpf-peer sa-cache summary vrf } | Displays the Multicast Source Discovery |
| show ip multicast [interface limit mpls redundancy vrf] | Displays global multicast information. |
| show ip pim all-vrfs { tunnel } | Display all VRFs. |
| show ip pim autorp | Display global auto-RP information. |
| show ip pim boundary [<i>type number</i>] | Displays boundary information. |
| show ip pim bsr-router | Display bootstrap router information (ve |
| show ip pim interface [<i>type number</i>] [count detail df stats] | Displays information about interfaces co |
| show ip pim neighbor [<i>type number</i>] | Lists the PIM neighbors discovered by th |
| show ip pim mdt [bgp] | Displays multicast tunnel information. |
| show ip pim rp [<i>group-name</i> <i>group-address</i>] | Displays the RP routers associated with a images. |
| show ip pim rp-hash [<i>group-name</i> <i>group-address</i>] | Displays the RP to be chosen based upon |
| show ip pim tunnel [<i>tunnel</i> <i>verbose</i>] | Displays the registered tunnels. |
| show ip pim vrf <i>name</i> | Displays VPN routing and forwarding in |

| Command | Purpose |
|--|---|
| <code>show ip rpf {source-address name}</code> | Displays how the device is doing Reverse-P routing table, or static mroutes). Command parameters include: <ul style="list-style-type: none"> • <i>Host name or IP address</i>—IP name or • Select—Group-based VRF select infor • vrf—Selects VPN Routing/Forwarding |
| <code>show ip sap [group “session-name” detail]</code> | Displays the Session Announcement Proto Command parameters include: <ul style="list-style-type: none"> • <i>A.B.C.D</i>—IP group address. • <i>WORD</i>—Session name (in double quot • detail—Session details. |

Configuration Examples for Basic IP Multicast Routing

This section provides configuration examples for Basic IP Multicast Routing.

Example: Configuring an IP Multicast Boundary

This example shows how to set up a boundary for all administratively-scoped addresses:

```
(config)# access-list 1 deny 239.0.0.0 0.255.255.255
(config)# access-list 1 permit 224.0.0.0 15.255.255.255
(config)# interface gigabitethernet1/0/1
(config-if)# ip multicast boundary 1
```

Example: Responding to mriinfo Requests

The software answers mriinfo requests sent by mrouted systems and Cisco routers and multilayer . The software returns information about neighbors through DVMRP tunnels and all the routed interfaces. This information includes the metric (always set to 1), the configured TTL threshold, the status of the interface, and various flags. You can also use the **mriinfo** privileged EXEC command to query the router or itself, as in this example:

```
# mriinfo
171.69.214.27 (mm1-7kd.cisco.com) [version cisco 11.1] [flags: PMS]:
171.69.214.27 -> 171.69.214.26 (mm1-r7kb.cisco.com) [1/0/pim/querier]
171.69.214.27 -> 171.69.214.25 (mm1-45a.cisco.com) [1/0/pim/querier]
171.69.214.33 -> 171.69.214.34 (mm1-45c.cisco.com) [1/0/pim]
171.69.214.137 -> 0.0.0.0 [1/0/pim/querier/down/leaf]
171.69.214.203 -> 0.0.0.0 [1/0/pim/querier/down/leaf]
171.69.214.18 -> 171.69.214.20 (mm1-45e.cisco.com) [1/0/pim]
171.69.214.18 -> 171.69.214.19 (mm1-45c.cisco.com) [1/0/pim]
```

```
171.69.214.18 -> 171.69.214.17 (mm1-45a.cisco.com) [1/0/pim]
```

Additional References for Basic IP Multicast Routing

Related Documents

| Related Topic | Document Title |
|--|---|
| For complete syntax and usage information for the commands used in this chapter. | See the IP Multicast Routing Commands section of the <i>Command Reference (Catalyst 9200 Series Switches)</i> |

Feature History for Basic IP Multicast Routing

This table provides release and related information for the features explained in this module.

These features are available in all the releases subsequent to the one they were introduced in, unless noted otherwise.

| Release | Feature | Feature Information |
|-------------------------------|----------------------|--|
| Cisco IOS XE Fuji 16.9.2 | IP Multicast Routing | IP Multicast is an efficient way to use network resources, especially for bandwidth-intensive services such as audio and video. IP multicast routing enables a host (source) to send packets to a group of hosts (receivers) anywhere within the IP network by using a special form of IP address called the IP multicast group address. |
| Cisco IOS XE Cupertino 17.9.1 | IP Multicast Routing | This feature was implemented on C9200CX-12P-2X2G, C9200CX-8P-2X2G, and C9200CX-12T-2X2G models of the Cisco Catalyst 9200CX Series Switches, which were introduced in this release. |

Use the Cisco Feature Navigator to find information about platform and software image support. To access Cisco Feature Navigator, go to <http://www.cisco.com/go/cfn>.



CHAPTER 3

Configuring Multicast Routing over GRE Tunnel

- [Prerequisites for Configuring Multicast Routing over GRE Tunnel, on page 37](#)
- [Restrictions for Configuring Multicast Routing over GRE Tunnel, on page 37](#)
- [Information About Multicast Routing over GRE Tunnel, on page 38](#)
- [How to Configure Multicast Routing over GRE Tunnel, on page 38](#)
- [Additional References for Multicast Routing over GRE Tunnel, on page 41](#)
- [Feature History for Multicast Routing over GRE Tunnel, on page 41](#)

Prerequisites for Configuring Multicast Routing over GRE Tunnel

Before configuring multicast routing over GRE, you should be familiar with the concepts of IP Multicast Routing Technology and GRE Tunneling.

Restrictions for Configuring Multicast Routing over GRE Tunnel

The following are the restrictions for configuring multicast routing over GRE tunnel:

- IPv6 multicast over GRE tunnel is not supported.
- The total number of supported multicast routes (mroutes) is 1024, across all tunnels.
- Bidirectional PIM is not supported.
- Multicast routing should be configured on the first hop router (FHR), the rendezvous point (RP) and the last hop router (LHR) to support multicast over the GRE tunnel.
- On Catalyst 9000 Series Switches, the tunnel source can be a loopback, physical, or L3 EtherChannel interface.
- No feature interactions such as IPSec, ACL, Tunnel counters, Crypto support, Fragmentation, Cisco Discovery Protocol (CDP), QoS, GRE keepalive, Multipoint GRE, etc. are supported on the GRE Tunnel.
- Tunnel source cannot be a subinterface.

Information About Multicast Routing over GRE Tunnel

This chapter describes how to configure a Generic Route Encapsulation (GRE) tunnel to tunnel IP multicast packets between non-IP multicast areas. The benefit is that IP multicast traffic can be sent from a source to a multicast group, over an area where IP multicast is not supported. Multicast Routing over GRE Tunnel supports sparse mode and pim-ssm mode; and supports static RP and auto-RP. See Rendezvous Point and Auto-RP for information on configuring static RP and auto-RP.

Benefits of Tunneling to Connect Non-IP Multicast Areas

- If the path between a source and a group member (destination) does not support IP multicast, a tunnel between them can transport IP multicast packets.

How to Configure Multicast Routing over GRE Tunnel

This section provides steps for configuring multicast routing over GRE tunnel.

Configuring a GRE Tunnel to Connect Non-IP Multicast Areas

You can configure a GRE tunnel to transport IP multicast packets between a source and destination that are connected by a medium that does not support multicast routing.

Procedure

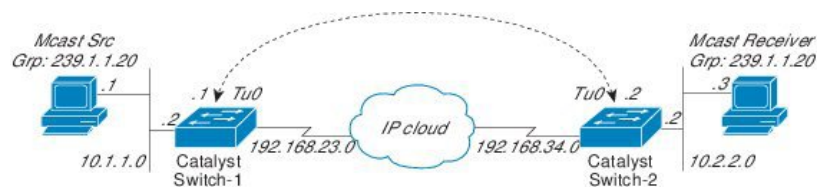
| | Command or Action | Purpose |
|---------------|--|---|
| Step 1 | enable Example: Device> enable | Enables privileged EXEC mode. • Enter your password if prompted. |
| Step 2 | configure terminal Example: Device# configure terminal | Enters global configuration mode. |
| Step 3 | ip multicast-routing Example: Device(config)# ip multicast-routing | Enables IP multicast routing. |
| Step 4 | interface tunnel <i>number</i> Example: Device(config)# interface tunnel 0 | Enters tunnel interface configuration mode. |

| | Command or Action | Purpose |
|----------------|--|---|
| Step 5 | ip address ip_address subnet_mask Example: Device(config-if)# ip address 192.168.24.1 255.255.255.252 | Configures IP address and IP subnet. |
| Step 6 | ip pim sparse-mode Example: Device(config-if)# ip pim sparse-mode | Enables sparse mode of operation of Protocol Independent Multicast (PIM) on the tunnel interface with one of the following mode of operation: |
| Step 7 | tunnel source { ip-address interface-name } Example: Device(config-if)# tunnel source 100.1.1.1 | Configures the tunnel source. |
| Step 8 | tunnel destination { hostname ip-address } Example: Device(config-if)# tunnel destination 100.1.1.3 | Configures the tunnel destination. |
| Step 9 | end Example: Device(config-if)# end | Ends the current configuration session and returns to privileged EXEC mode. |
| Step 10 | show interface type number Example: Device# show interface tunnel 0 | Displays tunnel interface information. |

Tunneling to Connect Non-IP Multicast Areas Example

The following example shows multicast-routing between a Catalyst switch through a GRE tunnel.

Figure 7: Tunnel Connecting Non-IP Multicast Areas



In the figure above, the multicast source (10.1.1.1) is connected to Catalyst Switch-1 and is configured for multicast group 239.1.1.20. The multicast receiver (10.2.2.3) is connected to Catalyst Switch-2 and is configured to receive multicast packets for group 239.1.1.20. Separating Switch-1 and Switch-2 is an IP cloud, which is not configured for multicast routing.

A GRE tunnel is configured between Switch-1 to Switch-2 sourced with their loopback interfaces. Multicast-routing is enabled on Switch-1 and Switch-2. The **ip pim sparse-mode** command is configured on tunnel interfaces to support PIM in the sparse mode. Sparse mode configuration on the tunnel interfaces allows sparse-mode packets to be forwarded over the tunnel depending on rendezvous point (RP) configuration for the group.

Switch-1 Configuration:

```
Device(config)# ip multicast-routing
Device(config)# interface Loopback0 //Tunnel source interface
Device(config-if)# ip address 2.2.2.2 255.255.255.255

Device(config)# interface Tunnel 10 //Tunnel interface configured for PIM
traffic
Device(config-if)# ip address 192.168.24.1 255.255.255.252
Device(config-if)# ip pim sparse-mode
Device(config-if)# ip nhrp map 192.168.24.3 4.4.4.4 //NHRP may optionally be
configured to dynamically discover tunnel end points.
Device(config-if)# ip nhrp map multicast 4.4.4.4
Device(config-if)# ip nhrp network-id 1
Device(config-if)# ip nhrp nhs 192.168.24.3
Device(config-if)# tunnel source Loopback0
Device(config-if)# tunnel destination 4.4.4.4

Device(config)# interface GigabitEthernet 0/0/0 //Source interface
Device(config-if)# ip address 10.1.1.2 255.255.255.0
Device(config-if)# ip pim sparse-mode
```

Switch-2 Configuration:

```
Device(config)# ip multicast-routing
Device(config)# interface Loopback0 //Tunnel source interface
Device(config-if)# ip address 4.4.4.4 255.255.255.255

Device(config)# interface Tunnel 10 //Tunnel interface configured for PIM
traffic
Device(config-if)# ip address 192.168.24.2 255.255.255.252
Device(config-if)# ip nhrp map 192.168.24.4 2.2.2.2 //NHRP may optionally be
configured to dynamically discover tunnel end points.
Device(config-if)# ip nhrp map multicast 2.2.2.2
Device(config-if)# ip nhrp network-id 1
Device(config-if)# ip nhrp nhs 192.168.24.4
Device(config-if)# ip pim sparse-mode
Device(config-if)# tunnel source Loopback0
Device(config-if)# tunnel destination 2.2.2.2

Device(config)# interface GigabitEthernet 0/0/0 //Receiver interface
Device(config-if)# ip address 10.2.2.2 255.255.255.0
Device(config-if)# ip pim sparse-mode
```

Additional References for Multicast Routing over GRE Tunnel

Related Documents

| Related Topic | Document Title |
|--|---|
| For complete syntax and usage information for the commands used in this chapter. | See the IP Multicast Routing Commands section of the <i>Command Reference (Catalyst 9200 Series Switches)</i> . |

Feature History for Multicast Routing over GRE Tunnel

This table provides release and related information for the features explained in this module.

These features are available in all the releases subsequent to the one they were introduced in, unless noted otherwise.

| Release | Feature | Feature Information |
|-------------------------------|-----------------------------------|---|
| Cisco IOS XE Fuji 16.9.2 | Multicast Routing over GRE Tunnel | Multicast routing over GRE tunnel allows IP multicast traffic to be sent from a source to a multicast group, over an area where IP multicast is not supported |
| Cisco IOS XE Cupertino 17.7.1 | Multicast Routing over GRE Tunnel | This feature was implemented on supervisor modules C9400X-SUP-2 and C9400X-SUP-2XL, which were introduced in this release. |
| Cisco IOS XE Cupertino 17.9.1 | Multicast Routing over GRE Tunnel | This feature was implemented on C9200CX-12P-2X2G, C9200CX-8P-2X2G, and C9200CX-12T-2X2G models of the Cisco Catalyst 9200CX Series Switches, which were introduced in this release. |

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.



CHAPTER 4

Configuring IGMP

- [Prerequisites for IGMP and IGMP Snooping, on page 43](#)
- [Restrictions for IGMP and IGMP Snooping, on page 44](#)
- [Information about IGMP, on page 45](#)
- [Default IGMP Configuration, on page 55](#)
- [How to Configure IGMP, on page 57](#)
- [How to Configure IGMP Snooping, on page 72](#)
- [Monitoring IGMP, on page 88](#)
- [Configuration Examples for IGMP, on page 91](#)
- [Additional References for IGMP, on page 96](#)
- [Feature History for IGMP, on page 96](#)

Prerequisites for IGMP and IGMP Snooping

Prerequisites for IGMP Snooping

Observe these guidelines when configuring the IGMP snooping querier:

- Configure the VLAN in global configuration mode.
- Configure an IP address on the VLAN interface. When enabled, the IGMP snooping querier uses the IP address as the query source address.
- If there is no IP address configured on the VLAN interface, the IGMP snooping querier tries to use the configured global IP address for the IGMP querier. If there is no global IP address specified, the IGMP querier tries to use the VLAN device virtual interface (SVI) IP address (if one exists). If there is no SVI IP address, the device uses the first available IP address configured on the device. The first IP address available appears in the output of the **show ip interface** privileged EXEC command. The IGMP snooping querier does not generate an IGMP general query if it cannot find an available IP address on the device.
- The IGMP snooping querier supports IGMP Versions 1 and 2.
- When administratively enabled, the IGMP snooping querier moves to the nonquerier state if it detects the presence of a multicast router in the network.
- When it is administratively enabled, the IGMP snooping querier moves to the operationally disabled state under these conditions:

- IGMP snooping is disabled in the VLAN.
- PIM is enabled on the SVI of the corresponding VLAN.

Restrictions for IGMP and IGMP Snooping

Restrictions for Configuring IGMP

The following are the restrictions for configuring IGMP:

- The device supports IGMP Versions 1, 2 , and 3.



Note For IGMP Version 3, only IGMP Version 3 BISS (Basic IGMPv3 Snooping Support) is supported.

- IGMP Version 3 uses new membership report messages that might not be correctly recognized by older IGMP snooping devices.
- IGMPv3 can operate with both ISM and SSM. In ISM, both exclude and include mode reports are applicable. In SSM, only include mode reports are accepted by the last-hop router. Exclude mode reports are ignored.
- Use ACLs to designate a specified port only as a multicast host port and not as a multicast router port. Multicast router control-packets received on this port are dropped.

Restrictions for IGMP Snooping

The following are the restrictions for IGMP snooping:

- The device supports IGMPv3 snooping based only on the destination multicast IP address. It does not support snooping based on a source IP address or proxy report.
- IGMPv3 join and leave messages are not supported on the devices running IGMP filtering or Multicast VLAN registration (MVR).
- IGMP report suppression is supported only when the multicast query has IGMPv1 and IGMPv2 reports. This feature is not supported when the query includes IGMPv3 reports.
- The IGMP configurable leave time is only supported on hosts running IGMP Version 2. IGMP version 2 is the default version for the device.

The actual leave latency in the network is usually the configured leave time. However, the leave time might vary around the configured time, depending on real-time CPU load conditions, network delays and the amount of traffic sent through the interface.

- The IGMP throttling action restriction can be applied only to Layer 2 ports. You can use **ip igmp max-groups action replace** interface configuration command on a logical EtherChannel interface but cannot use it on ports that belong to an EtherChannel port group.

When the maximum group limitation is set to the default (no maximum), entering the **ip igmp max-groups action {deny | replace}** command has no effect.

If you configure the throttling action and set the maximum group limitation after an interface has added multicast entries to the forwarding table, the forwarding-table entries are either aged out or removed, depending on the throttling action.

Information about IGMP

Role of the Internet Group Management Protocol

IGMP is used to dynamically register individual hosts in a multicast group on a particular LAN. Enabling PIM on an interface also enables IGMP. IGMP provides a means to automatically control and limit the flow of multicast traffic throughout your network with the use of special multicast queriers and hosts.

- A querier is a network device, such as a router, that sends query messages to discover which network devices are members of a given multicast group.
- A host is a receiver, including routers, that sends report messages (in response to query messages) to inform the querier of a host membership. Hosts use IGMP messages to join and leave multicast groups.

Hosts identify group memberships by sending IGMP messages to their local multicast device. Under IGMP, devices listen to IGMP messages and periodically send out queries to discover which groups are active or inactive on a particular subnet.

IGMP Multicast Addresses

IP multicast traffic uses group addresses, which are Class D IP addresses. The high-order four bits of a Class D address are 1110. Therefore, host group addresses can be in the range 224.0.0.0 to 239.255.255.255.

Multicast addresses in the range 224.0.0.0 to 224.0.0.255 are reserved for use by routing protocols and other network control traffic. The address 224.0.0.0 is guaranteed not to be assigned to any group.

IGMP packets are transmitted using IP multicast group addresses as follows:

- IGMP general queries are destined to the address 224.0.0.1 (all systems on a subnet).
- IGMP group-specific queries are destined to the group IP address for which the device is querying.
- IGMP group membership reports are destined to the group IP address for which the device is reporting.
- IGMPv2 leave-group messages are destined to the address 224.0.0.2 (all devices on a subnet).
- IGMPv3 membership reports are destined to the address 224.0.0.22; all IGMPv3-capable multicast devices must listen to this address.

IGMP Versions

The device supports IGMP version 1, IGMP version 2, and IGMP version 3. These versions are interoperable on the device. For example, if IGMP snooping is enabled and the querier's version is IGMPv2, and the device receives an IGMPv3 report from a host, then the device can forward the IGMPv3 report to the multicast router.

An IGMPv3 device can receive messages from and forward messages to a device running the Source Specific Multicast (SSM) feature.

IGMP Version 1

IGMP version 1 (IGMPv1) primarily uses a query-response model that enables the multicast router and multilayer device to find which multicast groups are active (have one or more hosts interested in a multicast group) on the local subnet. IGMPv1 has other processes that enable a host to join and leave a multicast group. For more information, see RFC 1112.

IGMP Version 2

IGMPv2 extends IGMP functionality by providing such features as the IGMP leave process to reduce leave latency, group-specific queries, and an explicit maximum query response time. IGMPv2 also adds the capability for routers to elect the IGMP querier without depending on the multicast protocol to perform this task. For more information, see RFC 2236.



Note IGMP version 2 is the default version for the device.

IGMP Version 3

The device supports IGMP version 3.

An IGMPv3 device supports Basic IGMPv3 Snooping Support (BISS), which includes support for the snooping features on IGMPv1 and IGMPv2 switches and for IGMPv3 membership report messages. BISS constrains the flooding of multicast traffic when your network includes IGMPv3 hosts. It constrains traffic to approximately the same set of ports as the IGMP snooping feature on IGMPv2 or IGMPv1 hosts.

An IGMPv3 device can receive messages from and forward messages to a device running the Source Specific Multicast (SSM) feature.

IGMPv3 Host Signaling

IGMPv3 is the third version of the IETF standards track protocol in which hosts signal membership to last-hop devices of multicast groups. IGMPv3 introduces the ability for hosts to signal group membership that allows filtering capabilities with respect to sources. A host can signal either that it wants to receive traffic from all sources sending to a group except for some specific sources (a mode called EXCLUDE) or that it wants to receive traffic only from some specific sources sending to the group (a mode called INCLUDE).

IGMPv3 can operate with both ISM and SSM. In ISM, both EXCLUDE and INCLUDE mode reports are accepted by the last-hop router. In SSM, only INCLUDE mode reports are accepted by the last-hop router.

IGMP Versions Differences

There are three versions of IGMP, as defined by Request for Comments (RFC) documents of the Internet Engineering Task Force (IETF). IGMPv2 improves over IGMPv1 by adding the ability for a host to signal desire to leave a multicast group and IGMPv3 improves over IGMPv2 mainly by adding the ability to listen to multicast originating from a set of source IP addresses only.

Table 5: IGMP Versions

| IGMP Version | Description |
|--------------|--|
| IGMPv1 | Provides the basic query-response mechanism that allows the multicast device to determine which multicast groups are active and other processes that enable hosts to join and leave a multicast group. RFC 1112 defines the IGMPv1 host extensions for IP multicasting. |
| IGMPv2 | Extends IGMP, allowing such capabilities as the IGMP leave process, group-specific queries, and an explicit maximum response time field. IGMPv2 also adds the capability for devices to elect the IGMP querier without dependence on the multicast protocol to perform this task. RFC 2236 defines IGMPv2. |
| IGMPv3 | Provides for source filtering, which enables a multicast receiver host to signal to a device which groups it wants to receive multicast traffic from, and from which sources this traffic is expected. In addition, IGMPv3 supports the link local address 224.0.0.22, which is the destination IP address for IGMPv3 membership reports; all IGMPv3-capable multicast devices must listen to this address. RFC 3376 defines IGMPv3. |



Note By default, enabling a PIM on an interface enables IGMPv2 on that device. IGMPv2 was designed to be as backward compatible with IGMPv1 as possible. To accomplish this backward compatibility, RFC 2236 defined special interoperability rules. If your network contains legacy IGMPv1 hosts, you should be familiar with these operability rules. For more information about IGMPv1 and IGMPv2 interoperability, see RFC 2236, Internet Group Management Protocol, Version 2 .

Devices That Run IGMPv1

IGMPv1 devices send IGMP queries to the “all-hosts” multicast address of 224.0.0.1 to solicit multicast groups with active multicast receivers. The multicast receivers also can send IGMP reports to the device to notify it that they are interested in receiving a particular multicast stream. Hosts can send the report asynchronously or in response to the IGMP queries sent by the device. If more than one multicast receiver exists for the same multicast group, only one of these hosts sends an IGMP report message; the other hosts suppress their report messages.

In IGMPv1, there is no election of an IGMP querier. If more than one device on the segment exists, all the devices send periodic IGMP queries. IGMPv1 has no special mechanism by which the hosts can leave the group. If the hosts are no longer interested in receiving multicast packets for a particular group, they simply do not reply to the IGMP query packets sent from the device. The device continues sending query packets. If the device does not hear a response in three IGMP queries, the group times out and the device stops sending multicast packets on the segment for the group. If the host later wants to receive multicast packets after the timeout period, the host simply sends a new IGMP join to the device, and the device begins to forward the multicast packet again.

If there are multiple devices on a LAN, a designated router (DR) must be elected to avoid duplicating multicast traffic for connected hosts. PIM devices follow an election process to select a DR. The PIM device with the highest IP address becomes the DR.

The DR is responsible for the following tasks:

- Sending PIM register and PIM Join and Prune messages toward the rendezvous point (RP) to inform it about host group membership.
- Sending IGMP host-query messages.
- Sending host-query messages by default every 60 seconds in order to keep the IGMP overhead on hosts and networks very low.

Devices That Run IGMPv2

IGMPv2 improves the query messaging capabilities of IGMPv1.

The query and membership report messages in IGMPv2 are identical to the IGMPv1 messages with two exceptions:

- IGMPv2 query messages are broken into two categories: general queries (identical to IGMPv1 queries) and group-specific queries.
- IGMPv1 membership reports and IGMPv2 membership reports have different IGMP type codes.

IGMPv2 also enhances IGMP by providing support for the following capabilities:

- Querier election process--Provides the capability for IGMPv2 devices to elect the IGMP querier without having to rely on the multicast routing protocol to perform the process.
- Maximum Response Time field--A new field in query messages permits the IGMP querier to specify the maximum query-response time. This field permits the tuning of the query-response process to control response burstiness and to fine-tune leave latencies.
- Group-Specific Query messages--Permits the IGMP querier to perform the query operation on a specific group instead of all groups.
- Leave-Group messages--Provides hosts with a method of notifying devices on the network that they wish to leave the group.

Unlike IGMPv1, in which the DR and the IGMP querier are typically the same device, in IGMPv2 the two functions are decoupled. The DR and the IGMP querier are selected based on different criteria and may be different devices on the same subnet. The DR is the device with the highest IP address on the subnet, whereas the IGMP querier is the device with the lowest IP address.

Query messages are used to elect the IGMP querier as follows:

1. When IGMPv2 devices start, they each multicast a general query message to the all-systems group address of 224.0.0.1 with their interface address in the source IP address field of the message.
2. When an IGMPv2 device receives a general query message, the device compares the source IP address in the message with its own interface address. The device with the lowest IP address on the subnet is elected the IGMP querier.
3. All devices (excluding the querier) start the query timer, which is reset whenever a general query message is received from the IGMP querier. If the query timer expires, it is assumed that the IGMP querier has gone down, and the election process is performed again to elect a new IGMP querier.

By default, the timer is two times the query interval.

Devices Running IGMPv3

IGMPv3 adds support for source filtering, which enables a multicast receiver host to signal to a device which groups it wants to receive multicast traffic from, and from which sources this traffic is expected. This membership information enables the software to forward traffic only from those sources from which receivers requested the traffic.

IGMPv3 supports applications that explicitly signal sources from which they want to receive traffic. With IGMPv3, receivers signal membership to a multicast group in the following two modes:

- **INCLUDE mode**--In this mode, the receiver announces membership to a group and provides a list of IP addresses (the INCLUDE list) from which it wants to receive traffic.
- **EXCLUDE mode**--In this mode, the receiver announces membership to a group and provides a list of IP addresses (the EXCLUDE list) from which it does not want to receive traffic. In other words, the host wants to receive traffic only from sources whose IP addresses are not listed in the EXCLUDE list. To receive traffic from all sources, like in the case of the Internet Standard Multicast (ISM) service model, a host expresses EXCLUDE mode membership with an empty EXCLUDE list.

IGMPv3 is the industry-designated standard protocol for hosts to signal channel subscriptions in an SSM network environment. For SSM to rely on IGMPv3, IGMPv3 must be available in the network stack portion of the operating systems running on the last hop devices and hosts and be used by the applications running on those hosts.

In IGMPv3, hosts send their membership reports to 224.0.0.22; all IGMPv3 devices, therefore, must listen to this address. Hosts, however, do not listen or respond to 224.0.0.22; they only send their reports to that address. In addition, in IGMPv3, there is no membership report suppression because IGMPv3 hosts do not listen to the reports sent by other hosts. Therefore, when a general query is sent out, all hosts on the wire respond.

IGMP Join and Leave Process

IGMP Join Process

When a host wants to join a multicast group, the host sends one or more unsolicited membership reports for the multicast group it wants to join. The IGMP join process is the same for IGMPv1 and IGMPv2 hosts.

In IGMPv3, the join process for hosts proceeds as follows:

- When a host wants to join a group, it sends an IGMPv3 membership report to 224.0.0.22 with an empty EXCLUDE list.
- When a host wants to join a specific channel, it sends an IGMPv3 membership report to 224.0.0.22 with the address of the specific source included in the INCLUDE list.
- When a host wants to join a group excluding particular sources, it sends an IGMPv3 membership report to 224.0.0.22 excluding those sources in the EXCLUDE list.



Note If some IGMPv3 hosts on a LAN wish to exclude a source and others wish to include the source, then the device will send traffic for the source on the LAN (that is, inclusion trumps exclusion in this situation).

IGMP Leave Process

The method that hosts use to leave a group varies depending on the version of IGMP in operation.

IGMPv1 Leave Process

There is no leave-group message in IGMPv1 to notify the devices on the subnet that a host no longer wants to receive the multicast traffic from a specific group. The host simply stops processing traffic for the multicast group and ceases responding to IGMP queries with IGMP membership reports for the group. As a result, the only way IGMPv1 devices know that there are no longer any active receivers for a particular multicast group on a subnet is when the devices stop receiving membership reports. To facilitate this process, IGMPv1 devices associate a countdown timer with an IGMP group on a subnet. When a membership report is received for the group on the subnet, the timer is reset. For IGMPv1 devices, this timeout interval is typically three times the query interval (3 minutes). This timeout interval means that the device may continue to forward multicast traffic onto the subnet for up to 3 minutes after all hosts have left the multicast group.

IGMPv2 Leave Process

IGMPv2 incorporates a leave-group message that provides the means for a host to indicate that it wishes to stop receiving multicast traffic for a specific group. When an IGMPv2 host leaves a multicast group, if it was the last host to respond to a query with a membership report for that group, it sends a leave-group message to the all-devices multicast group (224.0.0.2).

IGMPv3 Leave Process

IGMPv3 enhances the leave process by introducing the capability for a host to stop receiving traffic from a particular group, source, or channel in IGMP by including or excluding sources, groups, or channels in IGMPv3 membership reports.

IGMP Snooping

Layer 2 can use IGMP snooping to constrain the flooding of multicast traffic by dynamically configuring Layer 2 interfaces so that multicast traffic is forwarded to only those interfaces associated with IP multicast devices. As the name implies, IGMP snooping requires the LAN device to snoop on the IGMP transmissions between the host and the router and to keep track of multicast groups and member ports. When the device receives an IGMP report from a host for a particular multicast group, the device adds the host port number to the forwarding table entry; when it receives an IGMP Leave Group message from a host, it removes the host port from the table entry. It also periodically deletes entries if it does not receive IGMP membership reports from the multicast clients.



Note For more information on IP multicast and IGMP, see RFC 1112 and RFC 2236.

The multicast router set on the active device sends out periodic general queries to all VLANs. All hosts interested in this multicast traffic send join requests and are added to the forwarding table entry. The device creates one entry per VLAN in the IGMP snooping IP multicast forwarding table for each group from which it receives an IGMP join request.

The device supports IP multicast group-based bridging, instead of MAC-addressed based groups. With multicast MAC address-based groups, if an IP address being configured translates (aliases) to a previously

configured MAC address or to any reserved multicast MAC addresses (in the range 224.0.0.xxx), the command fails. Because the device uses IP multicast groups, there are no address aliasing issues.

The IP multicast groups learned through IGMP snooping are dynamic. However, you can statically configure multicast groups by using the **ip igmp snooping vlan *vlan-id* static *ip_address* interface *interface-id*** global configuration command. If you specify group membership for a multicast group address statically, your setting supersedes any automatic manipulation by IGMP snooping. Multicast group membership lists can consist of both user-defined and IGMP snooping-learned settings.

You can configure an IGMP snooping querier to support IGMP snooping in subnets without multicast interfaces because the multicast traffic does not need to be routed.

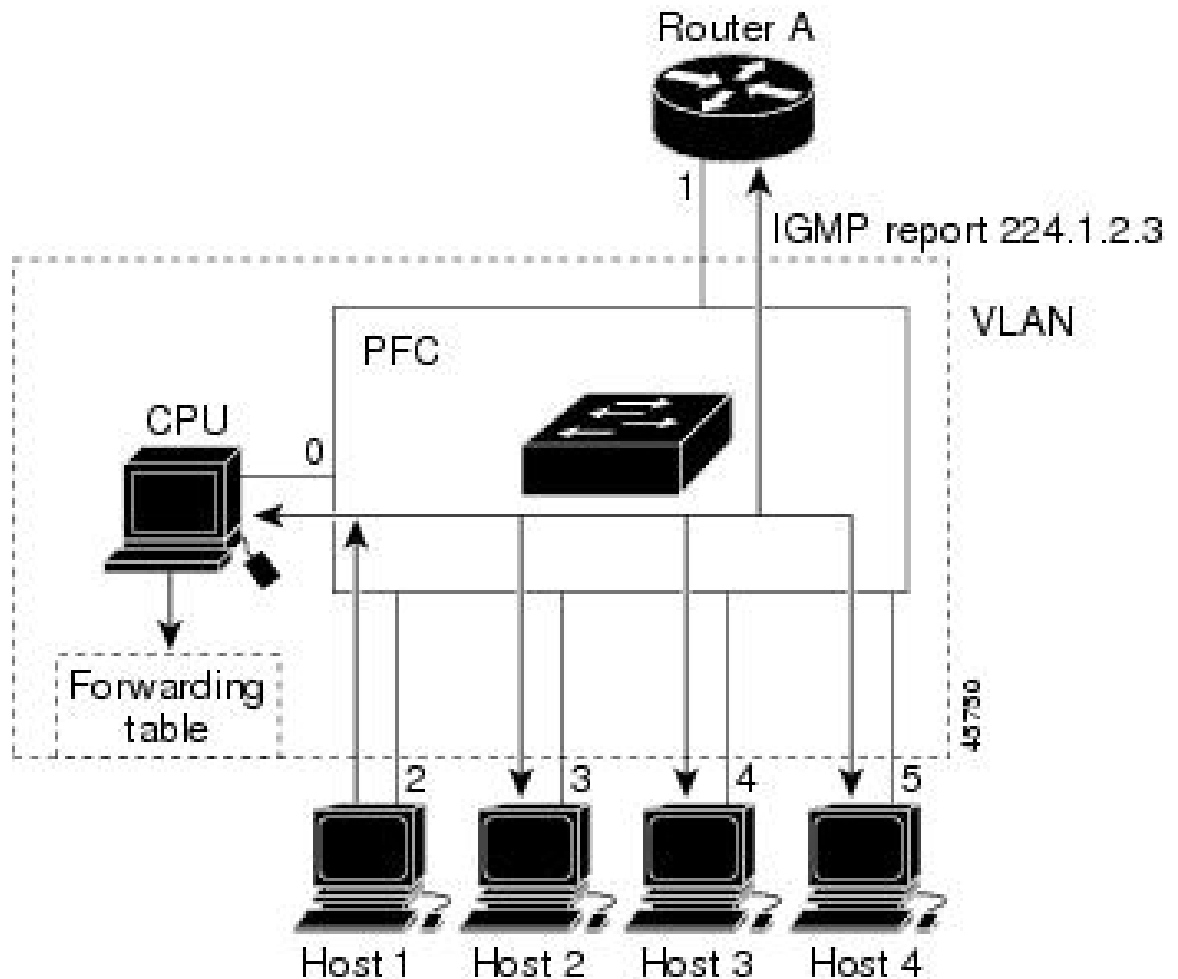
If a port spanning-tree, a port group, or a VLAN ID change occurs, the IGMP snooping-learned multicast groups from this port on the VLAN are deleted.

These sections describe IGMP snooping characteristics:

Joining a Multicast Group

Figure 8: Initial IGMP Join Message

When a host connected to the device wants to join an IP multicast group and it is an IGMP version 2 client, it sends an unsolicited IGMP join message, specifying the IP multicast group to join. Alternatively, when the device receives a general query from the router, it forwards the query to all ports in the VLAN. IGMP version 1 or version 2 hosts wanting to join the multicast group respond by sending a join message to the device. The device CPU creates a multicast forwarding-table entry for the group if it is not already present. The CPU also adds the interface where the join message was received to the forwarding-table entry. The host associated with that interface receives multicast traffic for that multicast group.



Router A sends a general query to the device, which forwards the query to ports 2 through 5, all of which are members of the same VLAN. Host 1 wants to join multicast group 224.1.2.3 and multicasts an IGMP membership report (IGMP join message) to the group. The device CPU uses the information in the IGMP report to set up a forwarding-table entry that includes the port numbers connected to Host 1 and to the router.

Table 6: IGMP Snooping Forwarding Table

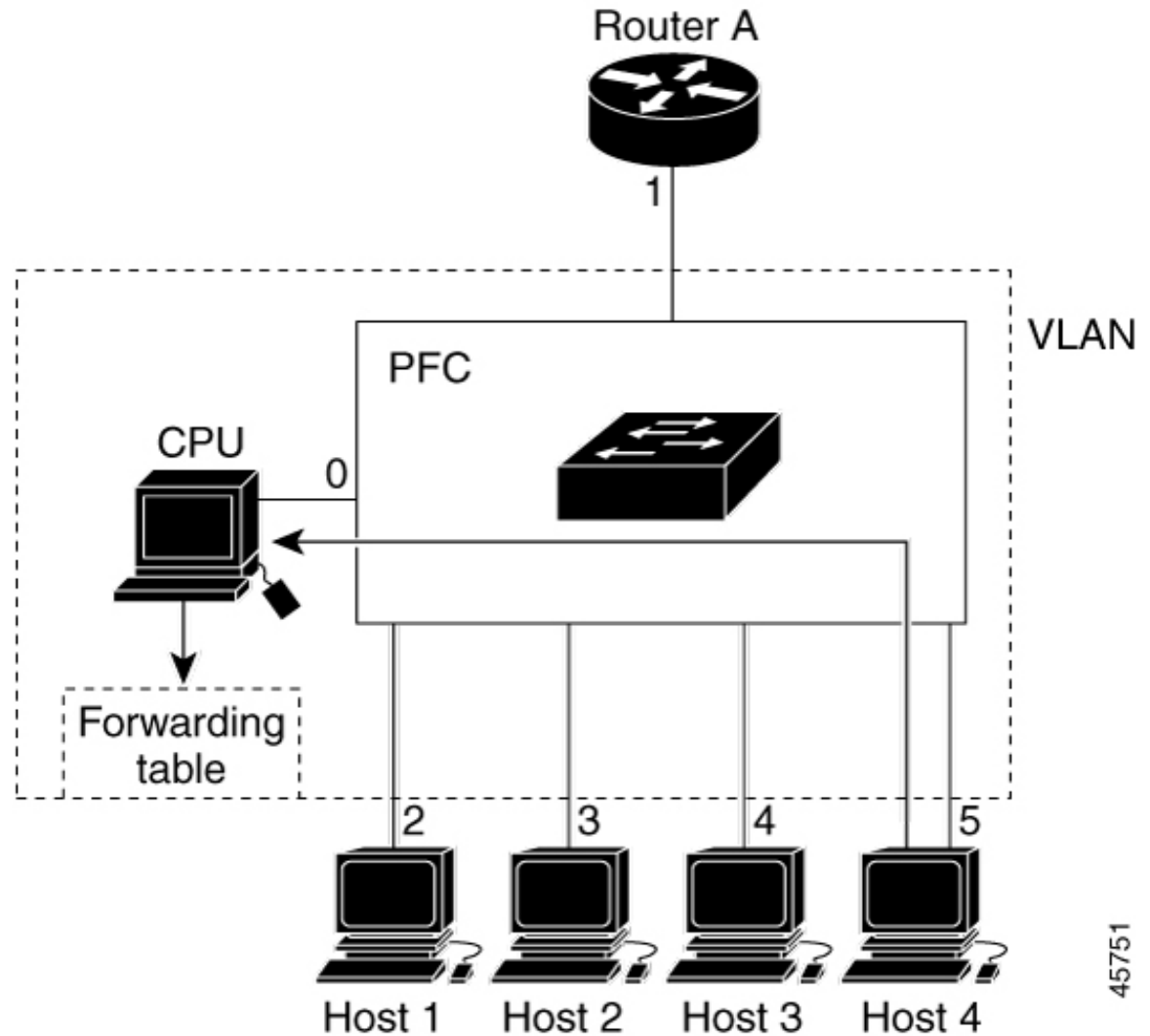
| Destination Address | Type of Packet | Ports |
|---------------------|----------------|-------|
| 224.1.2.3 | IGMP | 1, 2 |

The device hardware can distinguish IGMP information packets from other packets for the multicast group. The information in the table tells the switching engine to send frames addressed to the 224.1.2.3 multicast IP address that are not IGMP packets to the router and to the host that has joined the group.

Figure 9: Second Host Joining a Multicast Group

If another host (for example, Host 4) sends an unsolicited IGMP join message for the same group, the CPU receives that message and adds the port number of Host 4 to the forwarding table. Because the forwarding table directs IGMP messages only to the CPU, the message is not flooded to other ports on the device. Any

known multicast traffic is forwarded to the group and not to the CPU.



45751

Table 7: Updated IGMP Snooping Forwarding Table

| Destination Address | Type of Packet | Ports |
|---------------------|----------------|---------|
| 224.1.2.3 | IGMP | 1, 2, 5 |

Leaving a Multicast Group

The router sends periodic multicast general queries, and the device forwards these queries through all ports in the VLAN. Interested hosts respond to the queries. If at least one host in the VLAN wants to receive multicast traffic, the router continues forwarding the multicast traffic to the VLAN. The device forwards multicast group traffic only to those hosts listed in the forwarding table for that IP multicast group maintained by IGMP snooping.

When hosts want to leave a multicast group, they can silently leave, or they can send a leave message. When the device receives a leave message from a host, it sends a group-specific query to learn if any other devices

connected to that interface are interested in traffic for the specific multicast group. The device then updates the forwarding table for that MAC group so that only those hosts interested in receiving multicast traffic for the group are listed in the forwarding table. If the router receives no reports from a VLAN, it removes the group for the VLAN from its IGMP cache.

Immediate Leave

The device uses IGMP snooping Immediate Leave to remove from the forwarding table an interface that sends a leave message without the device sending group-specific queries to the interface. The VLAN interface is pruned from the multicast tree for the multicast group specified in the original leave message. Immediate Leave ensures optimal bandwidth management for all hosts on a switched network, even when multiple multicast groups are simultaneously in use.

Immediate Leave is only supported on IGMP version 2 hosts. IGMP version 2 is the default version for the device.



Note You should use the Immediate Leave feature only on VLANs where a single host is connected to each port. If Immediate Leave is enabled on VLANs where more than one host is connected to a port, some hosts may be dropped inadvertently.

IGMP Configurable-Leave Timer

You can configure the time that the device waits after sending a group-specific query to determine if hosts are still interested in a specific multicast group. The IGMP leave response time can be configured from 100 to 32767 milliseconds.

IGMP Report Suppression

IGMP report suppression is supported only when the multicast query has IGMPv1 and IGMPv2 reports. This feature is not supported when the query includes IGMPv3 reports.

The device uses IGMP report suppression to forward only one IGMP report per multicast router query to multicast devices. When IGMP report suppression is enabled (the default), the device sends the first IGMP report from all hosts for a group to all the multicast routers. The device does not send the remaining IGMP reports for the group to the multicast routers. This feature prevents duplicate reports from being sent to the multicast devices.

If the multicast router query includes requests only for IGMPv1 and IGMPv2 reports, the device forwards only the first IGMPv1 or IGMPv2 report from all hosts for a group to all the multicast routers.

If the multicast router query also includes requests for IGMPv3 reports, the device forwards all IGMPv1, IGMPv2, and IGMPv3 reports for a group to the multicast devices.

If you disable IGMP report suppression, all IGMP reports are forwarded to the multicast routers.

IGMP Snooping and Device Stacks

IGMP snooping functions across the device stack; that is, IGMP control information from one device is distributed to all devices in the stack. Regardless of the stack member through which IGMP multicast data enters the stack, the data reaches the hosts that have registered for that group.

If a device in the stack fails or is removed from the stack, only the members of the multicast group that are on that device will not receive the multicast data. All other members of a multicast group on other devices in

the stack continue to receive multicast data streams. However, multicast groups that are common for both Layer 2 and Layer 3 (IP multicast routing) might take longer to converge if the active device is removed.

IGMP Filtering and Throttling

In some environments, for example, metropolitan or multiple-dwelling unit (MDU) installations, you might want to control the set of multicast groups to which a user on a switch port can belong. You can control the distribution of multicast services, such as IP/TV, based on some type of subscription or service plan. You might also want to limit the number of multicast groups to which a user on a switch port can belong.

With the IGMP filtering feature, you can filter multicast joins on a per-port basis by configuring IP multicast profiles and associating them with individual switch ports. An IGMP profile can contain one or more multicast groups and specifies whether access to the group is permitted or denied. If an IGMP profile denying access to a multicast group is applied to a switch port, the IGMP join report requesting the stream of IP multicast traffic is dropped, and the port is not allowed to receive IP multicast traffic from that group. If the filtering action permits access to the multicast group, the IGMP report from the port is forwarded for normal processing. You can also set the maximum number of IGMP groups that a Layer 2 interface can join.

IGMP filtering controls only group-specific query and membership reports, including join and leave reports. It does not control general IGMP queries. IGMP filtering has no relationship with the function that directs the forwarding of IP multicast traffic. The filtering feature operates in the same manner whether CGMP or MVR is used to forward the multicast traffic.

IGMP filtering applies only to the dynamic learning of IP multicast group addresses, not static configuration.

With the IGMP throttling feature, you can set the maximum number of IGMP groups that a Layer 2 interface can join. If the maximum number of IGMP groups is set, the IGMP snooping forwarding table contains the maximum number of entries, and the interface receives an IGMP join report, you can configure an interface to drop the IGMP report or to replace the randomly selected multicast entry with the received IGMP report.



Note IGMPv3 join and leave messages are not supported on a device running IGMP filtering.

Default IGMP Configuration

This table displays the default IGMP configuration for the device.

Table 8: Default IGMP Configuration

| Feature | Default Setting |
|--|---|
| Multilayer device as a member of a multicast group | No group memberships are defined. |
| Access to multicast groups | All groups are allowed on an interface. |
| IGMP version | Version 2 on all interfaces. |
| IGMP host-query message interval | 60 seconds on all interfaces. |
| IGMP query timeout | 60 seconds on all interfaces. |
| IGMP maximum query response time | 10 seconds on all interfaces. |

| Feature | Default Setting |
|--|-----------------|
| Multilayer device as a statically connected member | Disabled. |

Default IGMP Snooping Configuration

This table displays the default IGMP snooping configuration for the device.

Table 9: Default IGMP Snooping Configuration

| Feature | Default Setting |
|------------------------------------|-------------------------------|
| IGMP snooping | Enabled globally and per VLAN |
| Multicast routers | None configured |
| IGMP snooping Immediate Leave | Disabled |
| Static groups | None configured |
| TCN ¹ flood query count | 2 |
| TCN query solicitation | Disabled |
| IGMP snooping querier | Disabled |
| IGMP report suppression | Enabled |

¹ (1) TCN = Topology Change Notification

Default IGMP Filtering and Throttling Configuration

This table displays the default IGMP filtering and throttling configuration for the device.

Table 10: Default IGMP Filtering Configuration

| Feature | Default Setting |
|------------------------------------|---|
| IGMP filters | None applied. |
| IGMP maximum number of IGMP groups | No maximum set. Note When the maximum number of groups is in the for table, the default IGMP throttling action is to deny report. |
| IGMP profiles | None defined. |
| IGMP profile action | Deny the range addresses. |

How to Configure IGMP

Configuring the Device as a Member of a Group

You can configure the device as a member of a multicast group and discover multicast reachability in a network. If all the multicast-capable routers and multilayer devices that you administer are members of a multicast group, pinging that group causes all of these devices to respond. The devices respond to ICMP echo-request packets addressed to a group of which they are members. Another example is the multicast trace-route tools provided in the software.



Caution Performing this procedure might impact the CPU performance because the CPU will receive all data traffic for the group address.

This procedure is optional.

Procedure

| | Command or Action | Purpose |
|---------------|---|---|
| Step 1 | enable Example: Device> enable | Enabled privileged EXEC mode. • Enter your password if prompted. |
| Step 2 | configure terminal Example: Device# configure terminal | Enters global configuration mode. |
| Step 3 | interface <i>interface-id</i> Example: Device(config)# interface GigabitEthernet 1/0/1 | Specifies the interface on which you want to enable multicast routing, and enters interface configuration mode. |
| Step 4 | ip igmp join-group <i>group-address</i> Example: Device(config-if)# ip igmp join-group 225.2.2.2 | Configures the device to join a multicast group. By default, no group memberships are defined. For <i>group-address</i> , specify the multicast IP address in dotted decimal notation. |
| Step 5 | end Example: Device(config-if)# end | Returns to privileged EXEC mode. |

| | Command or Action | Purpose |
|---------------|---|--|
| Step 6 | show ip igmp interface <i>[interface-id]</i> Example: <pre>Device# show ip igmp interface GigabitEthernet 1/0/1</pre> | Verifies your entries. |
| Step 7 | copy running-config startup-config Example: <pre>Device# copy running-config startup-config</pre> | (Optional) Saves your entries in the configuration file. |

Changing the IGMP Version

By default, the switch uses IGMP Version 2, which provides features such as the IGMP query timeout and the maximum query response time.

All systems on the subnet must support the same version. The switch does not automatically detect Version 1 systems and switch to Version 1. You can mix Version 1 and Version 2 hosts on the subnet because Version 2 routers or switches always work correctly with IGMPv1 hosts.

Configure the switch for Version 1 if your hosts do not support Version 2.

This procedure is optional.

Procedure

| | Command or Action | Purpose |
|---------------|---|--|
| Step 1 | enable Example: <pre>Device> enable</pre> | Enabled privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted. |
| Step 2 | configure terminal Example: <pre>Device# configure terminal</pre> | Enters global configuration mode. |
| Step 3 | interface <i>interface-id</i> Example: <pre>Device(config)# interface gigabitethernet 1/0/1</pre> | Specifies the interface to be configured, and enters the interface configuration mode. |

| | Command or Action | Purpose |
|---------------|---|--|
| Step 4 | ip igmp version {1 2 3 } Example: <pre>Device(config-if)# ip igmp version 2</pre> | Specifies the IGMP version that the switch uses. Note If you change to Version 1, you cannot configure the ip igmp query-interval or the ip igmp query-max-response-time interface configuration commands. To return to the default setting, use the no ip igmp version interface configuration command. |
| Step 5 | end Example: <pre>Device(config-if)# end</pre> | Returns to privileged EXEC mode. |
| Step 6 | show ip igmp interface [interface-id] Example: <pre>Device# show ip igmp interface</pre> | Verifies your entries. |
| Step 7 | copy running-config startup-config Example: <pre>Device# copy running-config startup-config</pre> | (Optional) Saves your entries in the configuration file. |

Modifying the IGMP Host-Query Message Interval

The device periodically sends IGMP host-query messages to discover which multicast groups are present on attached networks. These messages are sent to the all-hosts multicast group (224.0.0.1) with a time-to-live (TTL) of 1. The device sends host-query messages to refresh its knowledge of memberships present on the network. If, after some number of queries, the software discovers that no local hosts are members of a multicast group, the software stops forwarding multicast packets to the local network from remote origins for that group and sends a prune message upstream toward the source.

The device elects a PIM designated router (DR) for the LAN (subnet). The designated router is responsible for sending IGMP host-query messages to all hosts on the LAN. In sparse mode, the designated router also sends PIM register and PIM join messages toward the RP router. With IGMPv2, the DR is the router or multilayer device with the highest IP address. With IGMPv1, the DR is elected according to the multicast routing protocol that runs on the LAN.

This procedure is optional.

Procedure

| | Command or Action | Purpose |
|---------------|--|--|
| Step 1 | enable Example: Device> enable | Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted. |
| Step 2 | configure terminal Example: Device# configure terminal | Enters global configuration mode. |
| Step 3 | interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet 1/0/1 | Specifies the interface on which you want to enable multicast routing, and enters interface configuration mode. |
| Step 4 | ip igmp query-interval <i>seconds</i> Example: Device(config-if)# ip igmp query-interval 75 | Configures the frequency at which the designated router sends IGMP host-query messages. By default, the designated router sends IGMP host-query messages every 60 seconds to keep the IGMP overhead very low on hosts and networks. |
| Step 5 | end Example: Device(config)# end | Returns to privileged EXEC mode. |
| Step 6 | show ip igmp interface [<i>interface-id</i>] Example: Device# show ip igmp interface | Displays |
| Step 7 | copy running-config startup-config Example: Device# copy running-config startup-config | (Optional) Saves your entries in the configuration file. |

Changing the Maximum Query Response Time for IGMPv2

If you are using IGMPv2, you can change the maximum query response time advertised in IGMP queries. The maximum query response time enables the device to quickly detect that there are no more directly connected group members on a LAN. Decreasing the value enables the device to prune groups faster.

This procedure is optional.

Procedure

| | Command or Action | Purpose |
|---------------|--|---|
| Step 1 | enable Example: Device> enable | Enabled privileged EXEC mode. <ul style="list-style-type: none">• Enter your password if prompted. |
| Step 2 | configure terminal Example: Device# configure terminal | Enters global configuration mode. |
| Step 3 | interface <i>interface-id</i> Example: Device(config)# interface GigabitEthernet 1/0/1 | Specifies the interface on which you want to enable multicast routing, and enters interface configuration mode. |
| Step 4 | ip igmp query-max-response-time <i>seconds</i> Example: Device(config-if)# ip igmp query-max-response-time 15 | Changes the maximum query response time advertised in IGMP queries. The default is 10 seconds. The range is 1 to 25. |
| Step 5 | end Example: Device(config-if)# end | Returns to privileged EXEC mode. |
| Step 6 | show ip igmp interface [<i>interface-id</i>] Example: Device# show ip igmp interface | Verifies your entries. |
| Step 7 | copy running-config startup-config Example: Device# copy running-config | (Optional) Saves your entries in the configuration file. |

| | Command or Action | Purpose |
|--|-----------------------------|---------|
| | <code>startup-config</code> | |

Configuring the Device as a Statically Connected Member

At various times, either there is not a group member on a network segment or a host that cannot report its group membership by using IGMP. However, you may want multicast traffic to be sent to that network segment. The following commands are used to pull multicast traffic down to a network segment:

- **ip igmp join-group**—The device accepts the multicast packets in addition to forwarding them. Accepting the multicast packets prevents the device from fast switching.
- **ip igmp static-group**—The device does not accept the packets itself, but only forwards them. This method enables fast switching. The outgoing interface appears in the IGMP cache, but the device itself is not a member, as evidenced by lack of an L (local) flag in the multicast route entry.

This procedure is optional.

Procedure

| | Command or Action | Purpose |
|---------------|---|---|
| Step 1 | enable Example: Device> <code>enable</code> | Enabled privileged EXEC mode. • Enter your password if prompted. |
| Step 2 | configure terminal Example: Device# <code>configure terminal</code> | Enters global configuration mode. |
| Step 3 | interface <i>interface-id</i> Example: Device(config)# <code>interface GigabitEthernet 1/0/1</code> | Specifies the interface on which you want to enable multicast routing, and enters interface configuration mode. |
| Step 4 | ip igmp static-group <i>group-address</i> Example: Device(config-if)# <code>ip igmp static-group 239.100.100.101</code> | Configures the device as a statically connected member of a group. By default, this feature is disabled. |
| Step 5 | end Example: Device(config-if)# <code>end</code> | Returns to privileged EXEC mode. |

| | Command or Action | Purpose |
|---------------|---|--|
| Step 6 | show ip igmp interface <i>[interface-id]</i> Example: <pre>Device# show ip igmp interface GigabitEthernet 1/0/1</pre> | Verifies your entries. |
| Step 7 | copy running-config startup-config Example: <pre>Device# copy running-config startup-config</pre> | (Optional) Saves your entries in the configuration file. |

Configuring IGMP Profiles

Follow these steps to create an IGMP profile:

This task is optional.

Procedure

| | Command or Action | Purpose |
|---------------|---|---|
| Step 1 | enable Example: <pre>Device> enable</pre> | Enabled privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted. |
| Step 2 | configure terminal Example: <pre>Device# configure terminal</pre> | Enters global configuration mode. |
| Step 3 | ip igmp profile <i>profile number</i> Example: <pre>Device(config)# ip igmp profile 3</pre> | Assigns a number to the profile you are configuring, and enters IGMP profile configuration mode. The profile number range is 1 to 4294967295. When you are in IGMP profile configuration mode, you can create the profile by using these commands: <ul style="list-style-type: none"> • deny—Specifies that matching addresses are denied; this is the default. • exit—Exits from igmp-profile configuration mode. • no—Negates a command or returns to its defaults. • permit—Specifies that matching addresses are permitted. |

| | Command or Action | Purpose |
|---------------|---|---|
| | | <ul style="list-style-type: none"> range—Specifies a range of IP addresses for the profile. You can enter a single IP address or a range with a start and an end address. <p>The default for the device is to have no IGMP profiles configured.</p> <p>Note To delete a profile, use the no ip igmp profile <i>profile number</i> global configuration command.</p> |
| Step 4 | <p>permit deny</p> <p>Example:</p> <pre>Device(config-igmp-profile) # permit</pre> | <p>(Optional) Sets the action to permit or deny access to the IP multicast address. If no action is configured, the default for the profile is to deny access.</p> |
| Step 5 | <p>range <i>ip multicast address</i></p> <p>Example:</p> <pre>Device(config-igmp-profile) # range 229.9.9.0</pre> | <p>Enters the IP multicast address or range of IP multicast addresses to which access is being controlled. If entering a range, enter the low IP multicast address, a space, and the high IP multicast address.</p> <p>You can use the range command multiple times to enter multiple addresses or ranges of addresses.</p> <p>Note To delete an IP multicast address or range of IP multicast addresses, use the no range <i>ip multicast address</i> IGMP profile configuration command.</p> |
| Step 6 | <p>end</p> <p>Example:</p> <pre>Device(config-if) # end</pre> | <p>Returns to privileged EXEC mode.</p> |
| Step 7 | <p>show ip igmp profile <i>profile number</i></p> <p>Example:</p> <pre>Device# show ip igmp profile 3</pre> | <p>Verifies the profile configuration.</p> |
| Step 8 | <p>show running-config</p> <p>Example:</p> <pre>Device# show running-config</pre> | <p>Verifies your entries.</p> |

| | Command or Action | Purpose |
|---------------|---|--|
| Step 9 | copy running-config startup-config Example: <pre>Device# copy running-config startup-config</pre> | (Optional) Saves your entries in the configuration file. |

Applying IGMP Profiles

To control access as defined in an IGMP profile, you have to apply the profile to the appropriate interfaces. You can apply IGMP profiles only to Layer 2 access ports; you cannot apply IGMP profiles to routed ports or SVIs. You cannot apply profiles to ports that belong to an EtherChannel port group. You can apply a profile to multiple interfaces, but each interface can have only one profile applied to it.

Follow these steps to apply an IGMP profile to a switch port:

Procedure

| | Command or Action | Purpose |
|---------------|---|--|
| Step 1 | enable Example: <pre>Device> enable</pre> | Enabled privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted. |
| Step 2 | configure terminal Example: <pre>Device# configure terminal</pre> | Enters global configuration mode. |
| Step 3 | interface <i>interface-id</i> Example: <pre>Device(config)# interface GigabitEthernet 1/0/1</pre> | Specifies the physical interface, and enters interface configuration mode. The interface must be a Layer 2 port that does not belong to an EtherChannel port group. |
| Step 4 | ip igmp filter <i>profile number</i> Example: <pre>Device(config-if)# ip igmp filter 321</pre> | Applies the specified IGMP profile to the interface. The range is 1 to 4294967295. Note To remove a profile from an interface, use the no ip igmp filter <i>profile number</i> interface configuration command. |
| Step 5 | end Example: <pre>Device(config-if)# end</pre> | Returns to privileged EXEC mode. |

| | Command or Action | Purpose |
|---------------|---|--|
| Step 6 | show running-config Example: Device# <code>show running-config</code> | Verifies your entries. |
| Step 7 | copy running-config startup-config Example: Device# <code>copy running-config startup-config</code> | (Optional) Saves your entries in the configuration file. |

Setting the Maximum Number of IGMP Groups

Follow these steps to set the maximum number of IGMP groups that a Layer 2 interface can join:

Before you begin

This restriction can be applied to Layer 2 ports only; you cannot set a maximum number of IGMP groups on routed ports or SVIs. You also can use this command on a logical EtherChannel interface but cannot use it on ports that belong to an EtherChannel port group.

Procedure

| | Command or Action | Purpose |
|---------------|--|---|
| Step 1 | enable Example: Device> <code>enable</code> | Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted. |
| Step 2 | configure terminal Example: Device# <code>configure terminal</code> | Enters global configuration mode. |
| Step 3 | interface <i>interface-id</i> Example: Device(config)# <code>interface gigabitethernet1/0/2</code> | Specifies the interface to be configured, and enters interface configuration mode. The interface can be a Layer 2 port that does not belong to an EtherChannel group or a EtherChannel interface. |
| Step 4 | ip igmp max-groups <i>number</i> Example: Device(config-if)# <code>ip igmp max-groups 20</code> | Sets the maximum number of IGMP groups that the interface can join. The range is 0 to 4294967294. The default is to have no maximum set. |

| | Command or Action | Purpose |
|---------------|--|--|
| Step 5 | end Example: Device(config)# end | Returns to privileged EXEC mode. |
| Step 6 | show running-config interface <i>interface-id</i> Example: Device# show running-config interface gigabitethernet1/0/1 | Verifies your entries. |
| Step 7 | copy running-config startup-config Example: Device# copy running-config startup-config | (Optional) Saves your entries in the configuration file. |

Configuring the IGMP Throttling Action

After you set the maximum number of IGMP groups that a Layer 2 interface can join, you can configure an interface to replace the existing group with the new group for which the IGMP report was received.

Follow these steps to configure the throttling action when the maximum number of entries is in the forwarding table:

Procedure

| | Command or Action | Purpose |
|---------------|--|---|
| Step 1 | enable Example: Device> enable | Enabled privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted. |
| Step 2 | configure terminal Example: Device# configure terminal | Enters global configuration mode. |
| Step 3 | interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet1/0/1 | Specifies the physical interface to be configured, and enters interface configuration mode. The interface can be a Layer 2 port that does not belong to an EtherChannel group or an EtherChannel interface. The interface cannot be a trunk port. |

| | Command or Action | Purpose |
|--------|--|---|
| Step 4 | <p>ip igmp max-groups action {deny replace}</p> <p>Example:</p> <pre>Device(config-if)# ip igmp max-groups action replace</pre> | <p>When an interface receives an IGMP report and the maximum number of entries is in the forwarding table, specifies the action that the interface takes:</p> <ul style="list-style-type: none"> • deny—Drops the report. If you configure this throttling action, the entries that were previously in the forwarding table are not removed but are aged out. After these entries are aged out and the maximum number of entries is in the forwarding table, the device drops the next IGMP report received on the interface. • replace—Replaces the existing group with the new group for which the IGMP report was received. If you configure this throttling action, the entries that were previously in the forwarding table are removed. When the maximum number of entries is in the forwarding table, the device replaces a randomly selected entry with the received IGMP report. <p>To prevent the device from removing the forwarding-table entries, you can configure the IGMP throttling action before an interface adds entries to the forwarding table.</p> <p>Note To return to the default action of dropping the report, use the no ip igmp max-groups action interface configuration command.</p> |
| Step 5 | <p>end</p> <p>Example:</p> <pre>Device(config-if)# end</pre> | Returns to privileged EXEC mode. |
| Step 6 | <p>show running-config interface <i>interface-id</i></p> <p>Example:</p> <pre>Device# show running-config interface gigabitethernet1/0/1</pre> | Verifies your entries. |
| Step 7 | <p>copy running-config startup-config</p> <p>Example:</p> | (Optional) Saves your entries in the configuration file. |

| | Command or Action | Purpose |
|--|---|---------|
| | Device# <code>copy running-config startup-config</code> | |

Configuring the Device to Forward Multicast Traffic in the Absence of Directly Connected IGMP Hosts

Perform this optional task to configure the device to forward multicast traffic in the absence of directly connected IGMP hosts.

Procedure

| | Command or Action | Purpose |
|---------------|--|---|
| Step 1 | enable Example: Device> <code>enable</code> | Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted. |
| Step 2 | configure terminal Example: Device# <code>configure terminal</code> | Enters global configuration mode. |
| Step 3 | interface <i>type number</i> Example: Device(config)# <code>interface gigabitethernet 1/0/1</code> | Enters interface configuration mode. <ul style="list-style-type: none"> • For the <i>type</i> and <i>number</i> arguments, specify an interface that is connected to hosts. |
| Step 4 | Do one of the following: <ul style="list-style-type: none"> • ip igmp join-group <i>group-address</i> • ip igmp static-group <i>{* group-address [source source-address]}</i> Example: Device(config-if)# <code>ip igmp join-group 225.2.2.2</code> Example: Device(config-if)# <code>ip igmp static-group 225.2.2.2</code> | The first sample shows how to configure an interface on the device to join the specified group. With this method, the device accepts the multicast packets in addition to forwarding them. Accepting the multicast packets prevents the device from fast switching. The second example shows how to configure static group membership entries on an interface. With this method, the device does not accept the packets itself, but only forwards them. Hence, this method allows fast switching. The outgoing interface appears in the IGMP cache, but the device itself is not a member, as evidenced by lack of an “L” (local) flag in the multicast route entry |

| | Command or Action | Purpose |
|---------------|---|---|
| Step 5 | end Example: Device# (config-if) # end | Returns to privileged EXEC mode. |
| Step 6 | show ip igmp interface [<i>interface-type</i> <i>interface-number</i>] Example: Device# show ip igmp interface | (Optional) Displays multicast-related information about an interface. |

Controlling Access to an SSM Network Using IGMP Extended Access Lists

Perform this optional task to control access to an SSM network by using an IGMP extended access list that filters SSM traffic based on source address, group address, or both.

Procedure

| | Command or Action | Purpose |
|---------------|--|---|
| Step 1 | enable Example: Device> enable | Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted. |
| Step 2 | configure terminal Example: Device# configure terminal | Enters global configuration mode. |
| Step 3 | ip multicast-routing [distributed] Example: Device (config)# ip multicast-routing distributed | Enables IP multicast routing. <ul style="list-style-type: none"> • The distributed keyword is required for IPv4 multicast.. |
| Step 4 | ip pim ssm { default range <i>access-list</i> } Example: Device (config)# ip pim ssm default | Configures SSM service. <ul style="list-style-type: none"> • The default keyword defines the SSM range access list as 232/8. • The range keyword specifies the standard IP access list number or name that defines the SSM range. |
| Step 5 | ip access-list extended <i>access-list</i> -name Example: | Specifies an extended named IP access list. |

| | Command or Action | Purpose |
|----------------|--|--|
| | Device(config)# ip access-list extended mygroup | |
| Step 6 | <p>deny igmp <i>source source-wildcard destination destination-wildcard [igmp-type] [precedence precedence] [tos tos] [log] [time-range time-range-name] [fragments]</i></p> <p>Example:</p> <pre>Device(config-ext-nacl)# deny igmp host 10.1.2.3 any</pre> | <p>(Optional) Filters the specified source address or group address from the IGMP report, thereby restricting hosts on a subnet from membership to the (S, G) channel.</p> <ul style="list-style-type: none"> Repeat this step to restrict hosts on a subnet membership to other (S, G) channels. (These sources should be more specific than a subsequent permit statement because any sources or groups not specifically permitted are denied.) Remember that the access list ends in an implicit deny statement. This example shows how to create a deny statement that filters all groups for source 10.1.2.3, which effectively denies the source. |
| Step 7 | <p>permit igmp <i>source source-wildcard destination destination-wildcard [igmp-type] [precedence precedence] [tos tos] [log] [time-range time-range-name] [fragments]</i></p> <p>Example:</p> <pre>Device(config-ext-nacl)# permit igmp any any</pre> | <p>Allows a source address or group address in an IGMP report to pass the IP access list.</p> <ul style="list-style-type: none"> You must have at least one permit statement in an access list. Repeat this step to allow other sources to pass the IP access list. This example shows how to allow group membership to sources and groups not denied by prior deny statements. |
| Step 8 | <p>exit</p> <p>Example:</p> <pre>Device(config-ext-nacl)# exit</pre> | Exits the current configuration session and returns to global configuration mode. |
| Step 9 | <p>interface type number</p> <p>Example:</p> <pre>Device(config)# interface ethernet 0</pre> | Selects an interface that is connected to hosts on which IGMPv3 can be enabled. |
| Step 10 | <p>ip igmp access-group <i>access-list</i></p> <p>Example:</p> <pre>Device(config-if)# ip igmp access-group mygroup</pre> | Applies the specified access list to IGMP reports. |

| | Command or Action | Purpose |
|----------------|--|--|
| Step 11 | ip pim sparse-mode Example: Device(config-if)# ip pim sparse-mode | Enables PIM-SM on the interface. Note You must use sparse mode. |
| Step 12 | Repeat Steps 1 through 11 on all interfaces that require access control of SSM channel membership. | -- |
| Step 13 | ip igmp version 3 Example: Device(config-if)# ip igmp version 3 | Enables IGMPv3 on this interface. The default version of IGMP is IGMP version 2. Version 3 is required by SSM. |
| Step 14 | Repeat Step 13 on all host-facing interfaces. | -- |
| Step 15 | end Example: Device(config-if)# end | Returns to privileged EXEC mode. |

How to Configure IGMP Snooping

Enabling IGMP Snooping

Procedure

| | Command or Action | Purpose |
|---------------|---|---|
| Step 1 | enable Example: Device> enable | Enables privileged EXEC mode. • Enter your password if prompted. |
| Step 2 | configure terminal Example: Device# configure terminal | Enters global configuration mode. |
| Step 3 | ip igmp snooping Example: Device(config)# ip igmp snooping | Globally enables IGMP snooping after it has been disabled. |
| Step 4 | bridge-domain <i>bridge-id</i> Example: Device(config)# bridge-domain 100 | (Optional) Enters bridge domain configuration mode. |

| | Command or Action | Purpose |
|---------------|--|---|
| Step 5 | ip igmp snooping Example: Device(config-bdomain)# ip igmp snooping | (Optional) Enables IGMP snooping on the bridge domain interface being configured. <ul style="list-style-type: none"> • Required only if IGMP snooping was previously explicitly disabled on the specified bridge domain. |
| Step 6 | end Example: Device(config-bdomain)# end | Returns to privileged EXEC mode. |

Enabling or Disabling IGMP Snooping on a VLAN Interface

Follow these steps to enable IGMP snooping on a VLAN interface:

Procedure

| | Command or Action | Purpose |
|---------------|---|--|
| Step 1 | enable Example: Device> enable | Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted. |
| Step 2 | configure terminal Example: Device# configure terminal | Enters global configuration mode. |
| Step 3 | ip igmp snooping vlan <i>vlan-id</i> Example: Device(config)# ip igmp snooping vlan 7 | Enables IGMP snooping on the VLAN interface. The VLAN ID range is 1 to 1001 and 1006 to 4094. <p>IGMP snooping must be globally enabled before you can enable VLAN snooping.</p> <p>Note To disable IGMP snooping on a VLAN interface, use the no ip igmp snooping vlan <i>vlan-id</i> global configuration command for the specified VLAN number.</p> |
| Step 4 | end Example: Device(config)# end | Returns to privileged EXEC mode. |

| | Command or Action | Purpose |
|---------------|---|--|
| Step 5 | copy running-config startup-config Example: Device# copy running-config startup-config | (Optional) Saves your entries in the configuration file. |

Setting the Snooping Method

Multicast-capable router ports are added to the forwarding table for every Layer 2 multicast entry. The device learns of the ports through one of these methods:

- Snooping on IGMP queries, Protocol-Independent Multicast (PIM) packets
- Statically connecting to a multicast router port using the **ip igmp snooping mrouter** global configuration command

Beginning in privileged EXEC mode, follow these steps to alter the method in which a VLAN interface accesses a multicast router:

Procedure

| | Command or Action | Purpose |
|---------------|---|---|
| Step 1 | enable Example: Device> enable | Enables privileged EXEC mode. Enter your password, if prompted. |
| Step 2 | configure terminal Example: Device# configure terminal | Enters global configuration mode. |
| Step 3 | ip igmp snooping vlan <i>vlan-id</i> mrouter interface {GigabitEthernet Port-Channel TenGigabitEthernet} Example: Device(config)# ip igmp snooping vlan 1 mrouter interface GigabitEthernet1/0/3 | Enables IGMP snooping on a VLAN. The VLAN ID range is 1 to 1001 and 1006 to 4094. |
| Step 4 | end Example: Device(config)# end | Returns to privileged EXEC mode. |

| | Command or Action | Purpose |
|---------------|---|--|
| Step 5 | show ip igmp snooping Example: Device# <code>show ip igmp snooping</code> | Verifies the configuration. |
| Step 6 | copy running-config startup-config Example: Device# <code>copy running-config startup-config</code> | (Optional) Saves your entries in the configuration file. |

Configuring a Multicast Router Port

Perform these steps to add a multicast router port (enable a static connection to a multicast router) on the device.



Note Static connections to multicast routers are supported only on device ports.

Procedure

| | Command or Action | Purpose |
|---------------|--|--|
| Step 1 | enable Example: Device> <code>enable</code> | Enabled privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted. |
| Step 2 | configure terminal Example: Device# <code>configure terminal</code> | Enters global configuration mode. |
| Step 3 | ip igmp snooping vlan <i>vlan-id</i> mrouter interface <i>interface-id</i> Example: Device(config)# <code>ip igmp snooping vlan 5 mrouter interface GigabitEthernet 1/0/1</code> | Specifies the multicast router VLAN ID and the interface to the multicast router. <ul style="list-style-type: none"> • The VLAN ID range is 1 to 1001 and 1006 to 4094. • The interface can be a physical interface or a port channel. The port-channel range is 1 to 128. |

| | Command or Action | Purpose |
|---------------|--|---|
| | | Note To remove a multicast router port from the VLAN, use the no ip igmp snooping vlan <i>vlan-id</i> mrouter interface <i>interface-id</i> global configuration command. |
| Step 4 | end Example: Device(config-if) # end | Returns to privileged EXEC mode. |
| Step 5 | show ip igmp snooping mrouter [vlan <i>vlan-id</i>] Example: Device# show ip igmp snooping mrouter vlan 5 | Verifies that IGMP snooping is enabled on the VLAN interface. |
| Step 6 | copy running-config startup-config Example: Device# copy running-config startup-config | (Optional) Saves your entries in the configuration file. |

Configuring a Host Statically to Join a Group

Hosts or Layer 2 ports normally join multicast groups dynamically, but you can also statically configure a host on an interface.

Follow these steps to add a Layer 2 port as a member of a multicast group:

Procedure

| | Command or Action | Purpose |
|---------------|---|---|
| Step 1 | enable Example: Device> enable | Enabled privileged EXEC mode. • Enter your password if prompted. |
| Step 2 | configure terminal Example: Device# configure terminal | Enters global configuration mode. |
| Step 3 | ip igmp snooping vlan <i>vlan-id</i> static <i>ip_address</i> interface <i>interface-id</i> Example: | Statically configures a Layer 2 port as a member of a multicast group: |

| | Command or Action | Purpose |
|---------------|---|--|
| | <pre>Device(config)# ip igmp snooping vlan 105 static 230.0.0.1 interface gigabitethernet1/0/1</pre> | <ul style="list-style-type: none"> • <i>vlan-id</i> is the multicast group VLAN ID. The range is 1 to 1001 and 1006 to 4094. • <i>ip-address</i> is the group IP address. • <i>interface-id</i> is the member port. It can be a physical interface or a port channel (1 to 128). <p>Note To remove the Layer 2 port from the multicast group, use the no ip igmp snooping vlan <i>vlan-id</i> static <i>mac-address</i> interface <i>interface-id</i> global configuration command.</p> |
| Step 4 | <p>end</p> <p>Example:</p> <pre>Device(config-if)# end</pre> | Returns to privileged EXEC mode. |
| Step 5 | <p>show ip igmp snooping groups</p> <p>Example:</p> <pre>Device# show ip igmp snooping groups</pre> | Verifies the member port and the IP address. |
| Step 6 | <p>copy running-config startup-config</p> <p>Example:</p> <pre>Device# copy running-config startup-config</pre> | (Optional) Saves your entries in the configuration file. |

Enabling IGMP Immediate Leave

When you enable IGMP Immediate Leave, the device immediately removes a port when it detects an IGMP Version 2 leave message on that port. You should use the Immediate-Leave feature only when there is a single receiver present on every port in the VLAN.



Note Immediate Leave is supported only on IGMP Version 2 hosts. IGMP Version 2 is the default version for the device.

Procedure

| | Command or Action | Purpose |
|---------------|---|--|
| Step 1 | enable Example: Device> enable | Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted. |
| Step 2 | configure terminal Example: Device# configure terminal | Enters global configuration mode. |
| Step 3 | ip igmp snooping vlan <i>vlan-id</i> immediate-leave Example: Device(config)# ip igmp snooping vlan 21 immediate-leave | Enables IGMP Immediate Leave on the VLAN interface. Note To disable IGMP Immediate Leave on a VLAN, use the no ip igmp snooping vlan <i>vlan-id</i> immediate-leave global configuration command. |
| Step 4 | end Example: Device(config)# end | Returns to privileged EXEC mode. |
| Step 5 | show ip igmp snooping vlan <i>vlan-id</i> Example: Device# show ip igmp snooping vlan 21 | Verifies that Immediate Leave is enabled on the VLAN interface. |
| Step 6 | end Example: Device(config)# end | Returns to privileged EXEC mode. |

Configuring the IGMP Leave Timer

You can configure the leave time globally or on a per-VLAN basis. Follow these steps to enable the IGMP configurable-leave timer:

Procedure

| | Command or Action | Purpose |
|---------------|---|---|
| Step 1 | enable Example: Device> enable | Enabled privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted. |
| Step 2 | configure terminal Example: Device# configure terminal | Enters global configuration mode. |
| Step 3 | ip igmp snooping last-member-query-interval <i>time</i> Example: Device(config)# ip igmp snooping last-member-query-interval 1000 | Configures the IGMP leave timer globally. The range is 100 to 32767 milliseconds. The default leave time is 1000 milliseconds. Note To globally reset the IGMP leave timer to the default setting, use the no ip igmp snooping last-member-query-interval global configuration command. |
| Step 4 | ip igmp snooping vlan <i>vlan-id</i> last-member-query-interval <i>time</i> Example: Device(config)# ip igmp snooping vlan 210 last-member-query-interval 1000 | (Optional) Configures the IGMP leave time on the VLAN interface. The range is 100 to 32767 milliseconds. Note Configuring the leave time on a VLAN overrides the globally configured timer. Note To remove the configured IGMP leave-time setting from the specified VLAN, use the no ip igmp snooping vlan <i>vlan-id</i> last-member-query-interval global configuration command. |
| Step 5 | end Example: Device(config-if)# end | Returns to privileged EXEC mode. |
| Step 6 | show ip igmp snooping Example: Device# show ip igmp snooping | (Optional) Displays the configured IGMP leave time. |

| | Command or Action | Purpose |
|---------------|---|--|
| Step 7 | copy running-config startup-config Example: <pre>Device# copy running-config startup-config</pre> | (Optional) Saves your entries in the configuration file. |

Configuring the IGMP Robustness-Variable

Use the following procedure to configure the IGMP robustness variable on the device.

The robustness variable is the integer used by IGMP snooping during calculations for IGMP messages. The robustness variable provides fine tuning to allow for expected packet loss.

Procedure

| | Command or Action | Purpose |
|---------------|--|--|
| Step 1 | enable Example: <pre>Device> enable</pre> | Enabled privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted. |
| Step 2 | configure terminal Example: <pre>Device# configure terminal</pre> | Enters global configuration mode. |
| Step 3 | ip igmp snooping robustness-variable <i>count</i> Example: <pre>Device(config)# ip igmp snooping robustness-variable 3</pre> | Configures the IGMP robustness variable. The range is 1 to 3 times. The recommended value for the robustness variable is 2. Use this command to change the value of the robustness variable for IGMP snooping from the default (2) to a specified value. |
| Step 4 | ip igmp snooping vlan <i>vlan-id</i> robustness-variable <i>count</i> Example: <pre>Device(config)#ip igmp snooping vlan 100 robustness-variable 3</pre> | (Optional) Configures the IGMP robustness variable on the VLAN interface. The range is 1 to 3 times. The recommended value for the robustness variable is 2. Note Configuring the robustness variable count on a VLAN overrides the globally configured value. |
| Step 5 | end Example: | Returns to privileged EXEC mode. |

| | Command or Action | Purpose |
|---------------|---|--|
| | <code>Device(config-if)# end</code> | |
| Step 6 | show ip igmp snooping Example: <code>Device# show ip igmp snooping</code> | (Optional) Displays the configured IGMP robustness variable count. |
| Step 7 | copy running-config startup-config Example: <code>Device# copy running-config startup-config</code> | (Optional) Saves your entries in the configuration file. |

Configuring the IGMP Last Member Query Count

To configure the number of times the device sends IGMP group-specific or group-source-specific (with IGMP version 3) query messages in response to receiving a group-specific or group-source-specific leave message, use this command.

Procedure

| | Command or Action | Purpose |
|---------------|---|--|
| Step 1 | enable Example: <code>Device> enable</code> | Enabled privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted. |
| Step 2 | configure terminal Example: <code>Device# configure terminal</code> | Enters global configuration mode. |
| Step 3 | ip igmp snooping last-member-query-count <i>count</i> Example: <code>Device(config)# ip igmp snooping last-member-query-count 3</code> | Configures the IGMP last member query count. The range is 1 to 7 messages. The default is 2 messages. |
| Step 4 | ip igmp snooping vlan <i>vlan-id</i> last-member-query-count <i>count</i> Example: | (Optional) Configures the IGMP last member query count on the VLAN interface. The range is 1 to 7 messages. |

| | Command or Action | Purpose |
|---------------|---|--|
| | Device(config)# <code>ip igmp snooping vlan 100 last-member-query-count 3</code> | Note Configuring the last member query count on a VLAN overrides the globally configured timer. |
| Step 5 | end Example: Device(config-if)# <code>end</code> | Returns to privileged EXEC mode. |
| Step 6 | show ip igmp snooping Example: Device# <code>show ip igmp snooping</code> | (Optional) Displays the configured IGMP last member query count. |
| Step 7 | copy running-config startup-config Example: Device# <code>copy running-config startup-config</code> | (Optional) Saves your entries in the configuration file. |

Configuring TCN-Related Commands

Controlling the Multicast Flooding Time After a TCN Event

You can configure the number of general queries by which multicast data traffic is flooded after a topology change notification (TCN) event. If you set the TCN flood query count to 1 the flooding stops after receiving 1 general query. If you set the count to 7, the flooding continues until 7 general queries are received. Groups are relearned based on the general queries received during the TCN event.

Some examples of TCN events are when the client location is changed and the receiver is on same port that was blocked but is now forwarding, and when a port goes down without sending a leave message.

Follow these steps to configure the TCN flood query count:

Procedure

| | Command or Action | Purpose |
|---------------|---|---|
| Step 1 | enable Example: Device> <code>enable</code> | Enabled privileged EXEC mode. • Enter your password if prompted. |
| Step 2 | configure terminal Example: Device# <code>configure terminal</code> | Enters global configuration mode. |

| | Command or Action | Purpose |
|---------------|--|--|
| Step 3 | ip igmp snooping tcn flood query count <i>count</i> Example: <pre>Device(config)# ip igmp snooping tcn flood query count 3</pre> | Specifies the number of IGMP general queries for which the multicast traffic is flooded. The range is 1 to 10. The default, the flooding query count is 2. Note To return to the default flooding query count, use the no ip igmp snooping tcn flood query count global configuration command. |
| Step 4 | end Example: <pre>Device(config-if)# end</pre> | Returns to privileged EXEC mode. |
| Step 5 | show ip igmp snooping Example: <pre>Device# show ip igmp snooping</pre> | Verifies the TCN settings. |
| Step 6 | copy running-config startup-config Example: <pre>Device# copy running-config startup-config</pre> | (Optional) Saves your entries in the configuration file. |

Recovering from Flood Mode

When a topology change occurs, the spanning-tree root sends a special IGMP leave message (also known as global leave) with the group multicast address 0.0.0.0. However, you can enable the device to send the global leave message whether it is the spanning-tree root or not. When the router receives this special leave, it immediately sends general queries, which expedite the process of recovering from the flood mode during the TCN event. Leaves are always sent if the device is the spanning-tree root regardless of this configuration.

Follow these steps to enable sending of leave messages:

Procedure

| | Command or Action | Purpose |
|---------------|--|--|
| Step 1 | enable Example: <pre>Device> enable</pre> | Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted. |

| | Command or Action | Purpose |
|---------------|---|---|
| Step 2 | configure terminal Example: Device# <code>configure terminal</code> | Enters global configuration mode. |
| Step 3 | ip igmp snooping tcn query solicit Example: Device(config)# <code>ip igmp snooping tcn query solicit</code> | Sends an IGMP leave message (global leave) to speed the process of recovering from the flood mode caused during a TCN event. By default, query solicitation is disabled. Note To return to the default query solicitation, use the no ip igmp snooping tcn query solicit global configuration command. |
| Step 4 | end Example: Device(config)# <code>end</code> | Returns to privileged EXEC mode. |
| Step 5 | show ip igmp snooping Example: Device# <code>show ip igmp snooping</code> | Verifies the TCN settings. |
| Step 6 | copy running-config startup-config Example: Device# <code>copy running-config startup-config</code> | (Optional) Saves your entries in the configuration file. |

Disabling Multicast Flooding During a TCN Event

When the device receives a TCN, multicast traffic is flooded to all the ports until 2 general queries are received. If the device has many ports with attached hosts that are subscribed to different multicast groups, this flooding might exceed the capacity of the link and cause packet loss. Follow these steps to control TCN flooding:

Procedure

| | Command or Action | Purpose |
|---------------|---|---|
| Step 1 | enable Example: Device> <code>enable</code> | Enabled privileged EXEC mode. • Enter your password if prompted. |

| | Command or Action | Purpose |
|---------------|--|---|
| Step 2 | configure terminal Example: Device# <code>configure terminal</code> | Enters global configuration mode. |
| Step 3 | interface <i>interface-id</i> Example: Device(config)# <code>interface GigabitEthernet 1/0/1</code> | Specifies the interface to be configured, and enters interface configuration mode. |
| Step 4 | no ip igmp snooping tcn flood Example: Device(config-if)# <code>no ip igmp snooping tcn flood</code> | Disables the flooding of multicast traffic during a spanning-tree TCN event. By default, multicast flooding is enabled on an interface. Note To re-enable multicast flooding on an interface, use the ip igmp snooping tcn flood interface configuration command. |
| Step 5 | end Example: Device(config-if)# <code>end</code> | Returns to privileged EXEC mode. |
| Step 6 | show ip igmp snooping Example: Device# <code>show ip igmp snooping</code> | Verifies the TCN settings. |
| Step 7 | copy running-config startup-config Example: Device# <code>copy running-config startup-config</code> | (Optional) Saves your entries in the configuration file. |

Configuring the IGMP Snooping Querier

Follow these steps to enable the IGMP snooping querier feature in a VLAN:

Procedure

| | Command or Action | Purpose |
|---------------|---|--|
| Step 1 | enable Example: Device> enable | Enabled privileged EXEC mode. <ul style="list-style-type: none">• Enter your password if prompted. |
| Step 2 | configure terminal Example: Device# configure terminal | Enters global configuration mode. |
| Step 3 | ip igmp snooping querier Example: Device(config)# ip igmp snooping querier | Enables the IGMP snooping querier. |
| Step 4 | ip igmp snooping querier address ip_address Example: Device(config)# ip igmp snooping querier address 172.16.24.1 | (Optional) Specifies an IP address for the IGMP snooping querier. If you do not specify an IP address, the querier tries to use the global IP address configured for the IGMP querier. Note The IGMP snooping querier does not generate an IGMP general query if it cannot find an IP address on the device. |
| Step 5 | ip igmp snooping querier query-interval interval-count Example: Device(config)# ip igmp snooping querier query-interval 30 | (Optional) Sets the interval between IGMP queriers. The range is 1 to 18000 seconds. |
| Step 6 | ip igmp snooping querier tcn query [count count interval interval] Example: Device(config)# ip igmp snooping querier tcn query interval 20 | (Optional) Sets the time between Topology Change Notification (TCN) queries. The count range is 1 to 10. The interval range is 1 to 255 seconds. |
| Step 7 | ip igmp snooping querier timer expiry timeout Example: Device(config)# ip igmp snooping querier timer expiry 180 | (Optional) Sets the length of time until the IGMP querier expires. The range is 60 to 300 seconds. |

| | Command or Action | Purpose |
|----------------|--|---|
| Step 8 | ip igmp snooping querier version <i>version</i> Example: <pre>Device (config) # ip igmp snooping querier version 2</pre> | (Optional) Selects the IGMP version number that the querier feature uses. Select 1 or 2. |
| Step 9 | end Example: <pre>Device (config-if) # end</pre> | Returns to privileged EXEC mode. |
| Step 10 | show ip igmp snooping vlan <i>vlan-id</i> Example: <pre>Device# show ip igmp snooping vlan 30</pre> | (Optional) Verifies that the IGMP snooping querier is enabled on the VLAN interface. The VLAN ID range is 1 to 1001 and 1006 to 4094. |
| Step 11 | copy running-config startup-config Example: <pre>Device# copy running-config startup-config</pre> | (Optional) Saves your entries in the configuration file. |

Disabling IGMP Report Suppression

Follow these steps to disable IGMP report suppression:

Procedure

| | Command or Action | Purpose |
|---------------|---|---|
| Step 1 | enable Example: <pre>Device> enable</pre> | Enabled privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted. |
| Step 2 | configure terminal Example: <pre>Device# configure terminal</pre> | Enters global configuration mode. |
| Step 3 | no ip igmp snooping report-suppression Example: <pre>Device (config) # no ip igmp snooping report-suppression</pre> | Disables IGMP report suppression. When report suppression is disabled, all IGMP reports are forwarded to the multicast routers. IGMP report suppression is enabled by default. |

| | Command or Action | Purpose |
|---------------|---|---|
| | | When IGMP report suppression is enabled, the device forwards only one IGMP report per multicast router query. Note To re-enable IGMP report suppression, use the ip igmp snooping report-suppression global configuration command. |
| Step 4 | end Example: Device(config-if) # end | Returns to privileged EXEC mode. |
| Step 5 | show ip igmp snooping Example: Device# show ip igmp snooping | Verifies that IGMP report suppression is disabled. |
| Step 6 | copy running-config startup-config Example: Device# copy running-config startup-config | (Optional) Saves your entries in the configuration file. |

Monitoring IGMP

You can display specific statistics, such as the contents of IP routing tables, caches, and databases.



Note This release does not support per-route statistics.

You can display information to learn resource usage and solve network problems. You can also display information about node reachability and discover the routing path that packets of your device are taking through the network.

You can use any of the privileged EXEC commands in the following table to display various routing statistics.

Table 11: Commands for Displaying System and Network Statistics

| Command | Purpose |
|----------------------------|-----------------------------------|
| show ip igmp filter | Displays IGMP filter information. |

| Command | Purpose |
|---|--|
| <code>show ip igmp groups [type-number detail]</code> | Displays the multicast groups that are IGMP. |
| <code>show ip igmp interface [type number]</code> | Displays multicast-related information. |
| <code>show ip igmp membership [name/group address all tracked]</code> | Displays IGMP membership information. |
| <code>show ip igmp profile [profile_number]</code> | Displays IGMP profile information. |
| <code>show ip igmp ssm-mapping [hostname/IP address]</code> | Displays IGMP SSM mapping information. |
| <code>show ip igmp static-group {class-map [interface [type]]}</code> | Displays static group information. |
| <code>show ip igmp vrf</code> | Displays the selected VPN routing/forwarding information. Note The <code>show ip igmp vrf vrf-name</code> command displays the snooping information for the specified VRF. Use the <code>show ip igmp groups</code> command to see the multicast groups for the VRF. |

Monitoring IGMP Snooping Information

You can display IGMP snooping information for dynamically learned and statically configured router ports and VLAN interfaces. You can also display MAC address multicast entries for a VLAN configured for IGMP snooping.

Table 12: Commands for Displaying IGMP Snooping Information

| Command | Purpose |
|--|--|
| <code>show ip igmp snooping detail</code> | Displays the operational state information. |
| <code>show ip igmp snooping groups [count [dynamic [count] user [count]]]</code> | Displays multicast table information. <ul style="list-style-type: none"> • count—Displays the total number of actual entries. • dynamic—Displays entries learned dynamically. • user—Displays only the user-configured entries. |
| <code>show ip igmp snooping groups [count [vlan vlan-id [A.B.C.D count]]</code> | Displays multicast table information. <ul style="list-style-type: none"> • count—Displays the total number of entries. • vlan—Displays group information for the specified VLAN. |
| <code>show ip igmp snooping igmpv2-tracking</code> | Displays the IGMP snooping tracking information. Note This command displays information for wireless LANs and not for wired LANs. Use the <code>show ip igmp snooping</code> command to display information for wired LANs. |

| Command | Purpose |
|---|---|
| show ip igmp snooping groups <i>vlan</i> <i>vlan-id</i> [<i>ip_address</i> count dynamic [count] user [count]] | Displays multicast table information for <ul style="list-style-type: none"> • <i>vlan-id</i>—The VLAN ID range is 1 to 4094. • count—Displays the total number of actual entries. • dynamic—Displays entries learned dynamically. • <i>ip_address</i>—Displays characteristics for a specific IP address. • user—Displays only the user-configured entries. |
| show ip igmp snooping mrouter [vlan <i>vlan-id</i>] | Displays information on dynamically learned multicast routers. <p>Note When you enable IGMP snooping on a VLAN, a multicast router is connected to the VLAN.</p> (Optional) Enter vlan <i>vlan-id</i> to display information for a specific VLAN. |
| show ip igmp snooping querier [detail vlan <i>vlan-id</i>] | Displays information about the IP address of the querier and the number of query messages in the VLAN. <p>(Optional) Enter detail to display the details of the querier.</p> (Optional) Enter vlan <i>vlan-id</i> to display information for a specific VLAN. |
| show ip igmp snooping querier [vlan <i>vlan-id</i>] detail | Displays information about the IP address of the querier and the number of query messages in the VLAN and the configuration of the querier in the VLAN. |
| show ip igmp snooping [vlan <i>vlan-id</i> [detail]] | Displays the snooping configuration information for all interfaces on the device. <p>(Optional) Enter vlan <i>vlan-id</i> to display information for a specific VLAN. The range is 1001 and 1006 to 4094.</p> |

Monitoring IGMP Filtering and Throttling Configuration

You can display IGMP profile characteristics, and you can display the IGMP profile and maximum group configuration for all interfaces on the device or for a specified interface. You can also display the IGMP throttling configuration for all interfaces on the device or for a specified interface.

Table 13: Commands for Displaying IGMP Filtering and Throttling Configuration

| Command | Purpose |
|---|---|
| show ip igmp profile [<i>profile number</i>] | Displays the specified IGMP profile or all the IGMP profiles defined on the device. |

| Command | Purpose |
|--|---|
| <code>show running-config [interface <i>interface-id</i>]</code> | Displays the configuration of the specified interface. If no interface is specified, displays the configuration of all interfaces on the device. (If an interface is configured) the maximum number of IGMP groups an interface can belong and the IGMP profile of the interface. |

Configuration Examples for IGMP

Example: Configuring the Device as a Member of a Multicast Group

This example shows how to enable the device to join multicast group 255.2.2.2:

```
Device(config)# interface gigabitEthernet1/0/1
Device(config-if)# ip igmp join-group 255.2.2.2
Device(config-if)#
```

Example: Controlling Access to Multicast Groups

To limit the number of joins on the interface, configure the port for filter which associates with the IGMP profile.

```
Device# configure terminal
Device(config)# ip igmp profile 10
Device(config-igmp-profile)# ?

IGMP profile configuration commands:
deny matching addresses are denied
exit Exit from igmp profile configuration mode
no Negate a command or set its defaults
permit matching addresses are permitted
range add a range to the set

Device(config-igmp-profile)# range 172.16.5.1
Device(config-igmp-profile)# exit
Device(config)# interface gigabitEthernet 2/0/10
Device(config-if)# ip igmp filter 10
```

Examples: Configuring IGMP Snooping

This example shows how to enable a static connection to a multicast router:

```
Device# configure terminal
Device(config)# ip igmp snooping vlan 200 mrouter interface gigabitEthernet1/0/2
Device(config)# end
```

This example shows how to statically configure a host on a port:

```
Device# configure terminal
Device(config)# ip igmp snooping vlan 105 static 224.2.4.12 interface gigabitethernet1/0/1
Device(config)# end
```

This example shows how to enable IGMP Immediate Leave on VLAN 130:

```
Device# configure terminal
Device(config)# ip igmp snooping vlan 130 immediate-leave
Device(config)# end
```

This example shows how to set the IGMP snooping querier source address to 10.0.0.64:

```
Device# configure terminal
Device(config)# ip igmp snooping querier 10.0.0.64
Device(config)# end
```

This example shows how to set the IGMP snooping querier maximum response time to 25 seconds:

```
Device# configure terminal
Device(config)# ip igmp snooping querier query-interval 25
Device(config)# end
```

This example shows how to set the IGMP snooping querier timeout to 60 seconds:

```
Device# configure terminal
Device(config)# ip igmp snooping querier timer expiry 60
Device(config)# end
```

This example shows how to set the IGMP snooping querier feature to Version 2:

```
Device# configure terminal
Device(config)# no ip igmp snooping querier version 2
Device(config)# end
```

Example: Configuring IGMP Profiles

This example shows how to create IGMP profile 4 allowing access to the single IP multicast address and how to verify the configuration. If the action was to deny (the default), it would not appear in the **show ip igmp profile** output display.

```
Device(config)# ip igmp profile 4
Device(config-igmp-profile)# permit
Device(config-igmp-profile)# range 229.9.9.0
Device(config-igmp-profile)# end
Device# show ip igmp profile 4
IGMP Profile 4
  permit
  range 229.9.9.0 229.9.9.0
```

Example: Applying IGMP Profile

This example shows how to apply IGMP profile 4 to a port:

```
Device(config)# interface gigabitEthernet1/0/2
Device(config-if)# ip igmp filter 4
Device(config-if)# end
```

Example: Setting the Maximum Number of IGMP Groups

This example shows how to limit to 25 the number of IGMP groups that a port can join:

```
Device(config)# interface GigabitEthernet1/0/2
Device(config-if)# ip igmp max-groups 25
Device(config-if)# end
```

Example: Interface Configuration as a Routed Port

This example shows how to configure an interface on the device as a routed port. This configuration is required on the interface for several IP multicast routing configuration procedures that require running the **no switchport** command.

```
Device# configure terminal
Device(config)# interface GigabitEthernet1/0/9
Device(config-if)# description interface to be use as routed port
Device(config-if)# no switchport
Device(config-if)# ip address 10.20.20.1 255.255.255.0
Device(config-if)# ip pim sparse-mode
Device(config-if)# ip igmp join-group 224.1.2.3 source 15.15.15.2
Device(config-if)# end
Device# configure terminal
Device# show run interface gigabitEthernet 1/0/9

Current configuration : 166 bytes
!
interface GigabitEthernet1/0/9
 no switchport
 ip address 10.20.20.1 255.255.255.0
 ip pim sparse-mode
 ip igmp static-group 224.1.2.3 source 15.15.15.2
end
```

Example: Interface Configuration as an SVI

This example shows how to configure an interface on the device as an SVI. This configuration is required on the interface for several IP multicast routing configuration procedures that require running the **no switchport** command.

```
Device(config)# interface vlan 150
Device(config-if)# ip address 10.20.20.1 255.255.255.0
Device(config-if)# ip pim sparse-mode
Device(config-if)# ip igmp join-group 224.1.2.3 source 15.15.15.2
Device(config-if)# end
Device# configure terminal
Device(config)# ip igmp snooping vlan 20 static 224.1.2.3 interface gigabitEthernet 1/0/9
Device# show run interface vlan 150

Current configuration : 137 bytes
!
interface vlan 150
```

```
ip address 10.20.20.1 255.255.255.0
ip pim sparse-mode
ip igmp static-group 224.1.2.3 source 15.15.15.2
end
```

Example: Configuring the Device to Forward Multicast Traffic in the Absence of Directly Connected IGMP Hosts

The following example shows how to configure a device to forward multicast traffic in the absence of directly connected IGMP hosts using the **ip igmp join-group** command. With this method, the device accepts the multicast packets in addition to forwarding them. Accepting the multicast packets prevents the device from fast switching.

In this example, GigabitEthernet interface 1/0/1 on the device is configured to join the group 225.2.2.2:

```
interface GigabitEthernet1/0/1
ip igmp join-group 225.2.2.2
```

The following example shows how to configure a device to forward multicast traffic in the absence of directly connected IGMP hosts using the **ip igmp static-group** command. With this method, the device does not accept the packets itself, but only forwards them. Hence, this method allows fast switching. The outgoing interface appears in the IGMP cache, but the device itself is not a member, as evidenced by lack of an “L” (local) flag in the multicast route entry.

In this example, static group membership entries for group 225.2.2.2 are configured on Fast Ethernet interface 0/1/0:

```
interface GigabitEthernet1/0/1
ip igmp static-group 225.2.2.2
```

Controlling Access to an SSM Network Using IGMP Extended Access Lists

This section contains the following configuration examples for controlling access to an SSM network using IGMP extended access lists:



Note Keep in mind that access lists are very flexible: there are many combinations of permit and deny statements one could use in an access list to filter multicast traffic. The examples in this section simply provide a few examples of how it can be done.

Example: Denying All States for a Group G

The following example shows how to deny all states for a group G. In this example, Fast Ethernet interface 0/0/0 is configured to filter all sources for SSM group 232.2.2.2 in IGMPv3 reports, which effectively denies this group.

```
ip access-list extended test1
deny igmp any host 232.2.2.2
permit igmp any any
!
```



```
interface GigabitEthernet 1/0/1
 ip igmp access-group test1
```

Example: Denying All States for a Source S

The following example shows how to deny all states for a source S. In this example, Gigabit Ethernet interface 1/1/0 is configured to filter all groups for source 10.2.1.32 in IGMPv3 reports, which effectively denies this source.

```
ip access-list extended test2
 deny igmp host 10.2.1.32 any
 permit igmp any any
!
interface GigabitEthernet1/0/1
 ip igmp access-group test2
```

Example: Permitting All States for a Group G

The following example shows how to permit all states for a group G. In this example, Gigabit Ethernet interface 1/2/0 is configured to accept all sources for SSM group 232.1.1.10 in IGMPv3 reports, which effectively accepts this group altogether.

```
ip access-list extended test3
 permit igmp any host 232.1.1.10
!
interface GigabitEthernet 1/2/0
 ip igmp access-group test3
```

Example: Permitting All States for a Source S

The following example shows how to permit all states for a source S. In this example, Gigabit Ethernet interface 1/2 is configured to accept all groups for source 10.6.23.32 in IGMPv3 reports, which effectively accepts this source altogether.

```
ip access-list extended test4
 permit igmp host 10.6.23.32 any
!
interface GigabitEthernet1/2/0
 ip igmp access-group test4
```

Example: Filtering a Source S for a Group G

The following example shows how to filter a particular source S for a group G. In this example, Gigabit Ethernet interface 0/3/0 is configured to filter source 232.2.2.2 for SSM group 232.2.30.30 in IGMPv3 reports.

```
ip access-list extended test5
 deny igmp host 10.4.4.4 host 232.2.30.30
 permit igmp any any
!
interface GigabitEthernet0/3/0
 ip igmp access-group test5
```

Additional References for IGMP

Related Documents

| Related Topic | Document Title |
|--|---|
| For complete syntax and usage information for the commands used in this chapter. | See the IP Multicast Routing Commands section of the <i>Command Reference (Catalyst 9200 Series Switches)</i> |

Feature History for IGMP

This table provides release and related information for the features explained in this module.

These features are available in all the releases subsequent to the one they were introduced in, unless noted otherwise.

| Release | Feature | Feature Information |
|-------------------------------|---------|--|
| Cisco IOS XE Fuji 16.9.2 | IGMP | IGMP is used to dynamically register individual hosts in a multicast group on a particular LAN. Enabling PIM on an interface also enables IGMP. IGMP provides a means to automatically control and limit the flow of multicast traffic throughout your network with the use of special multicast queriers and hosts. |
| Cisco IOS XE Cupertino 17.9.1 | IGMP | This feature was implemented on C9200CX-12P-2X2G, C9200CX-8P-2X2G, and C9200CX-12T-2X2G models of the Cisco Catalyst 9200CX Series Switches, which were introduced in this release. |

Use the Cisco Feature Navigator to find information about platform and software image support. To access Cisco Feature Navigator, go to <http://www.cisco.com/go/cfn>.



CHAPTER 5

Configuring IGMP Proxy

- [Prerequisites for IGMP Proxy, on page 97](#)
- [Information About IGMP Proxy, on page 97](#)
- [How to Configure IGMP Proxy, on page 101](#)
- [Configuration Examples for IGMP Proxy, on page 107](#)
- [Additional References for IGMP Proxy, on page 108](#)
- [Feature History for IGMP Proxy, on page 109](#)

Prerequisites for IGMP Proxy

- All devices on the IGMP UDL have the same subnet address. If all devices on the UDL cannot have the same subnet address, the upstream device must be configured with secondary addresses to match all of the subnets to which the downstream devices are attached.
- IP multicast is enabled and the PIM interfaces are configured. When you are configuring PIM interfaces for IGMP proxy, use PIM sparse mode (PIM-SM) when the interface is operating in a sparse-mode region and you are running static RP, bootstrap (BSR), or Auto-RP with the Auto-RP listener capability.

Information About IGMP Proxy

IGMP Proxy

An IGMP proxy enables hosts in a unidirectional link routing (UDLR) environment that are not directly connected to a downstream router to join a multicast group sourced from an upstream network.

There are two methods of implementing IGMP Proxy:

- IGMP Proxy for a Single Upstream Interface
- IGMP Proxy for Multiple Upstream Interfaces

IGMP Proxy for a Single Upstream Interface

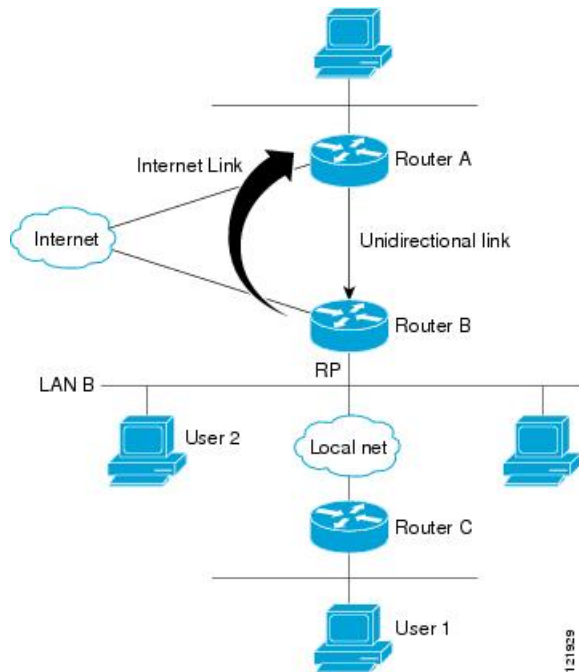
The [figure](#) below illustrates a sample topology that shows two UDLR scenarios:

- Traditional UDL routing scenario--A UDL device with directly connected receivers.
- IGMP proxy scenario--UDL device without directly connected receivers.

IGMP UDLs are needed on the upstream and downstream devices.



Note Although the following illustration and example uses routers in the configuration, any device (router or switch) can be used.



Scenario 1 - Traditional UDLR Scenario (UDL Device with Directly Connected Receivers)

For scenario 1, no IGMP proxy mechanism is needed. In this scenario, the following sequence of events occurs:

1. User 2 sends an IGMP membership report requesting interest in group G.
2. Router B receives the IGMP membership report, adds a forwarding entry for group G on LAN B, and proxies the IGMP report to Router A, which is the UDLR upstream device.
3. The IGMP report is then proxied across the Internet link.
4. Router A receives the IGMP proxy and maintains a forwarding entry on the unidirectional link.

Scenario 2 - IGMP Proxy Scenario (UDL Device without Directly Connected Receivers)

For scenario 2, the IGMP proxy mechanism is needed to enable hosts that are not directly connected to a downstream device to join a multicast group sourced from an upstream network. In this scenario, the following sequence of events occurs:

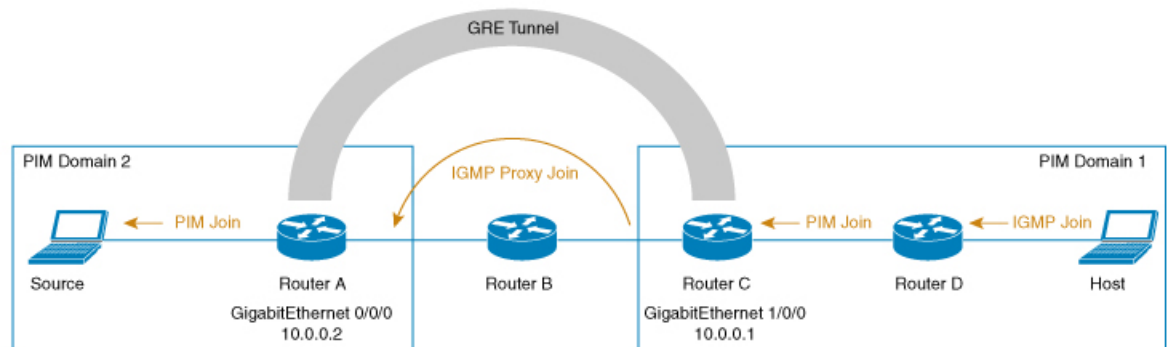
1. User 1 sends an IGMP membership report requesting interest in group G.

2. Router C sends a PIM Join message hop-by-hop to the RP (Router B).
3. Router B receives the PIM Join message and adds a forwarding entry for group G on LAN B.
4. Router B periodically checks its mroute table and proxies the IGMP membership report to its upstream UDL device across the Internet link.
5. Router A creates and maintains a forwarding entry on the unidirectional link (UDL).

In an enterprise network, it is desirable to be able to receive IP multicast traffic via satellite and forward the traffic throughout the network. With unidirectional link routing (UDLR) alone, scenario 2 would not be possible because receiving hosts must be directly connected to the downstream device, Router B. The IGMP proxy mechanism overcomes this limitation by creating an IGMP report for (*, G) entries in the multicast forwarding table. To make this scenario functional, therefore, you must enable IGMP report forwarding of proxied (*, G) multicast static route (mroute) entries (using the **ip igmp mroute-proxy** command) and enable the mroute proxy service (using the **ip igmp proxy-service** command) on interfaces leading to PIM-enabled networks with potential members.



Note Because PIM messages are not forwarded upstream, each downstream network and the upstream network have a separate domain.



Scenario 3 - IGMP Proxy Scenario without UDLR

For scenario 3, the IGMP proxy mechanism is used to enable hosts to receive traffic from an upstream network, without using a UDLR link. In this scenario, the following sequence of events occurs:

1. The host is in PIM Domain 1 and sends an IGMP membership report (a join request) to **Router D** requesting interest in group G. **Router D** converts the IGMP Join to a PIM join and sends it to **Router C**. This request should now be sent upstream, from **Router C** to **Router A**. The routers are in two different PIM domains (not PIM neighbors) and are connected through a GRE tunnel instead.
2. **Router C** converts the PIM join message to an IGMP proxy join so that it can be forwarded to the GRE tunnel endpoint, which is **Router A**.



Note IGMP proxy joins can be transferred across 1 hop only.

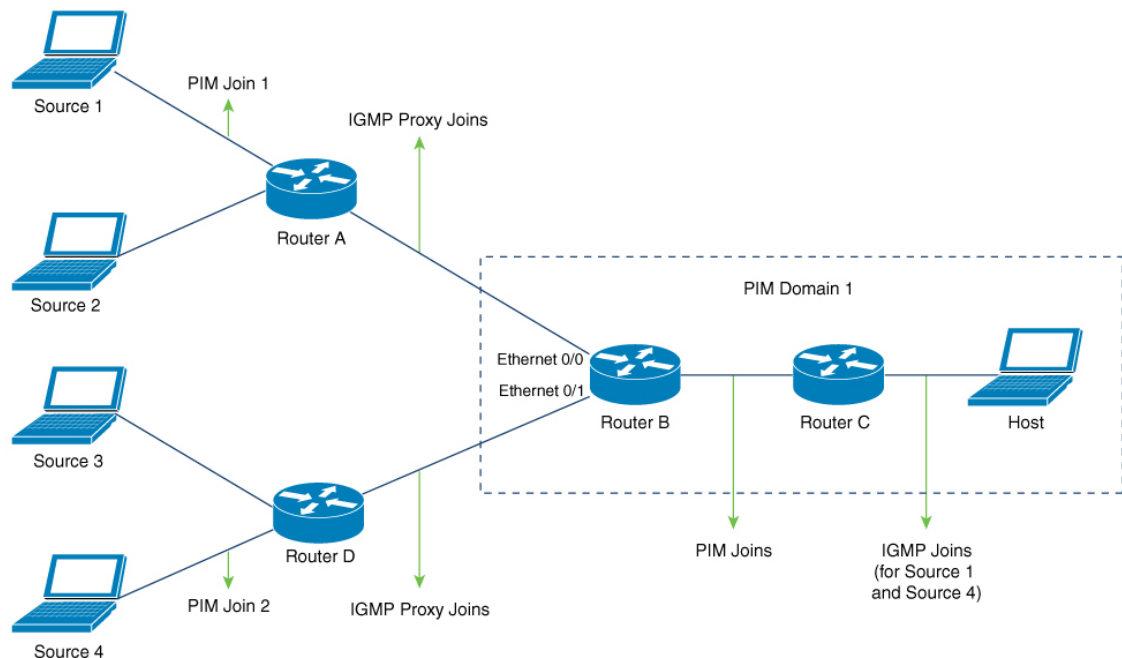
In the [figure](#) above, the GRE tunnel provides this single hop between Router C and Router A (bypassing Router B).

In the absence of a GRE tunnel, devices from different PIM domains must have directly (back-to-back) connected interfaces.

- After the IGMP proxy join reaches **Router A**, it is forwarded to the source device as a PIM join message.

IGMP Proxy for Multiple Upstream Interfaces

IGMP proxy also enables the user to request data from multiple upstream interfaces. You can implement IGMP proxy by this method if there are more number of upstream devices in the network. With this method, you can also implement IGMP proxy for a single upstream device as in any of the three scenarios described in the previous section.



In this method, IGMP proxy is used to enable the user to receive traffic from multiple upstream devices. The following sequence of events occurs:

- The host is in PIM Domain 1 and sends multiple IGMP membership reports (join requests) to **Router C** requesting interest in different groups. **Router C** converts the IGMP joins to PIM joins and sends them to **Router B**. These requests should be sent upstream, from **Router B** to **Router A**. The routers are in two different PIM domains (not PIM neighbors).
- Router B** converts the PIM join messages to IGMP proxy joins so that it can be forwarded to the upward interfaces.

3. A class-map is configured globally. This class-map describes the information about multicast groups. The IGMP proxy joins for different multicast groups are sent if the following conditions are met:
 - There is (*, G) or and (S, G) entry for that group.
 - (*, G) or the (S, G) entry has a NON-NULL OIF list.
4. During the IGMP proxy intervals, the IGMP proxy joins for different groups are sent through the respective upstream interfaces.
5. After the IGMP proxy join reaches **Router A**, it is forwarded to the different source devices as PIM join messages.

How to Configure IGMP Proxy

Configuring the Upstream UDL Device for IGMP UDLR

Perform this task to configure the upstream UDL device for IGMP UDLR.

Procedure

| | Command or Action | Purpose |
|---------------|--|---|
| Step 1 | enable Example: Device> enable | Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted. |
| Step 2 | configure terminal Example: Device# configure terminal | Enters global configuration mode. |
| Step 3 | interface <i>type number</i> Example: Device(config)# interface gigabitethernet 1/0/0 | Enters interface configuration mode. <ul style="list-style-type: none"> • For the <i>type</i> and <i>number</i> arguments, specify the interface to be used as the UDL on the upstream device. |
| Step 4 | ip igmp unidirectional-link Example: Device(config-if)# ip igmp unidirectional-link | Configures IGMP on the interface to be unidirectional for IGMP UDLR. |
| Step 5 | end Example: Device(config-if)# end | Ends the current configuration session and returns to privileged EXEC mode. |

Configuring the Downstream UDL Device for IGMP UDLR with IGMP Proxy Support

Perform this task to configure the downstream UDL device for IGMP UDLR with IGMP proxy support.

Procedure

| | Command or Action | Purpose |
|---------------|--|---|
| Step 1 | enable Example: <pre>Device> enable</pre> | Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted. |
| Step 2 | configure terminal Example: <pre>Device# configure terminal</pre> | Enters global configuration mode. |
| Step 3 | interface <i>type number</i> Example: <pre>Device(config)# interface gigabitethernet 0/0/0</pre> | Enters interface configuration mode. <ul style="list-style-type: none"> • For the <i>type</i> and <i>number</i> arguments, specify the interface to be used as the UDL on the downstream device for IGMP UDLR. |
| Step 4 | ip igmp unidirectional-link Example: <pre>Device(config-if)# ip igmp unidirectional-link</pre> | Configures IGMP on the interface to be unidirectional for IGMP UDLR. |
| Step 5 | exit Example: <pre>Device(config-if)# exit</pre> | Exits interface configuration mode and returns to global configuration mode. |
| Step 6 | interface <i>type number</i> Example: <pre>Device(config)# interface gigabitethernet 1/0/0</pre> | Enters interface configuration mode. <ul style="list-style-type: none"> • For the <i>type</i> and <i>number</i> arguments, select an interface that is facing the nondirectly connected hosts. |
| Step 7 | ip igmp mroute-proxy <i>type number</i> Example: <pre>Device(config-if)# ip igmp mroute-proxy loopback 0</pre> | Enables IGMP report forwarding of proxied (*, G) multicast static route (mroute) entries. <ul style="list-style-type: none"> • This step is performed to enable the forwarding of IGMP reports to a proxy service interface for all (*, G) forwarding entries in the multicast forwarding table. |

| | Command or Action | Purpose |
|----------------|---|---|
| | | <ul style="list-style-type: none"> In this example, the ip igmp mroute-proxy command is configured on Gigabit Ethernet interface 1/0/0 to request that IGMP reports be sent to loopback interface 0 for all groups in the mroute table that are forwarded to Gigabit Ethernet interface 1/0/0. |
| Step 8 | exit Example: <pre>Device(config-if)# exit</pre> | Exits interface configuration mode and returns to global configuration mode. |
| Step 9 | interface <i>type number</i> Example: <pre>Device(config)# interface loopback 0</pre> | Enters interface configuration mode for the specified interface. <ul style="list-style-type: none"> In this example, loopback interface 0 is specified. |
| Step 10 | ip igmp helper-address udl <i>interface-type interface-number</i> Example: <pre>Device(config-if)# ip igmp helper-address udl gigabitethernet 0/0/0</pre> | Configures IGMP helping for UDLR. <ul style="list-style-type: none"> This step allows the downstream device to helper IGMP reports received from hosts to an upstream device connected to a UDL associated with the interface specified for the <i>interface-type</i> and <i>interface-number</i> arguments. In the example topology, IGMP helping is configured over loopback interface 0 on the downstream device. Loopback interface 0, thus, is configured to helper IGMP reports from hosts to an upstream device connected to Gigabit Ethernet interface 0/0/0. |
| Step 11 | ip igmp proxy-service Example: <pre>Device(config-if)# ip igmp proxy-service</pre> | Enables the mroute proxy service. <ul style="list-style-type: none"> When the mroute proxy service is enabled, the device periodically checks the static mroute table for (*, G) forwarding entries that match interfaces configured with the ip igmp mroute-proxy command (see Step 7) based on the IGMP query interval. Where there is a match, one IGMP report is created and received on this interface. |

| | Command or Action | Purpose |
|----------------|---|---|
| | | <p>Note The ip igmp proxy-service command is intended to be used with the ip igmp helper-address (UDL) command.</p> <ul style="list-style-type: none"> In this example, the ip igmp proxy-service command is configured on loopback interface 0 to enable the forwarding of IGMP reports out the interface for all groups on interfaces registered through the ip igmp mroute-proxy command (see Step 7). |
| Step 12 | <p>end</p> <p>Example:</p> <pre>Device(config-if)# end</pre> | Ends the current configuration session and returns to privileged EXEC mode. |
| Step 13 | <p>show ip igmp interface</p> <p>Example:</p> <pre>Device# show ip igmp interface</pre> | (Optional) Displays multicast-related information about an interface. |
| Step 14 | <p>show ip igmp udldr</p> <p>Example:</p> <pre>Device# show ip igmp udldr</pre> | (Optional) Displays UDLR information for directly connected multicast groups on interfaces that have a UDL helper address configured. |

Configuring the Downstream Device for IGMP Proxy Join without UDLR

Perform this task to configure the downstream device for IGMP Proxy without UDLR.

(Referring to the [figure](#) above, all the steps are configured on *Router C*)

Procedure

| | Command or Action | Purpose |
|---------------|---|---|
| Step 1 | <p>enable</p> <p>Example:</p> <pre>Device > enable</pre> | Enables privileged EXEC mode. Enter your password if prompted. |
| Step 2 | <p>configure terminal</p> <p>Example:</p> <pre>Device# configure terminal</pre> | Enters global configuration mode. |
| Step 3 | <p>interface <i>type number</i></p> | Enters interface configuration mode. |

| | Command or Action | Purpose |
|---------------|--|---|
| | Example: <pre>Device(config)# interface gigabitethernet 1/0/0</pre> | For the <i>type</i> and <i>number</i> arguments, specify the interface that is facing the host. |
| Step 4 | ip igmp mroute-proxy type number Example: <pre>Device(config-if)# ip igmp mroute-proxy loopback 0</pre> | <p>Enables the forwarding of IGMP reports to the specified proxy service interface, for forwarding of all proxied (*, G) multicast static route (mroute) entries in the multicast forwarding table.</p> <p>In the step example, <i>loopback interface 0</i> is such a proxy service interface.</p> |
| Step 5 | exit Example: <pre>Device(config-if)# exit</pre> | Exits interface configuration mode and returns to global configuration mode. |
| Step 6 | interface type number Example: <pre>Device(config)# interface loopback 0</pre> | <p>Enters interface configuration mode for the specified proxy service interface.</p> <p>In the step example, <i>loopback interface 0</i> is specified.</p> |
| Step 7 | ip igmp helper-address ip-address Example: <pre>Device(config-if)# ip igmp helper-address 10.0.0.2</pre> | <p>Configures IGMP helper for IGMP proxy join.</p> <p>For the <i>ip-address</i> argument, specify the ip address of the upstream device to which the IGMP proxy join should reach.</p> <p>In the example topology, the IGMP helper is configured over loopback interface 0 on the downstream device (Router C).</p> <p>This command configures loopback interface 0 to help convert the PIM joins received from Router D to IGMP proxy joins and transfer them to the upstream device (Router A).</p> |
| Step 8 | ip igmp proxy-service Example: <pre>Device(config-if) ip igmp proxy-service</pre> | <p>Enables the mroute proxy service.</p> <p>When the mroute proxy service is enabled, the device periodically checks the static mroute table for (*, G) forwarding entries that match interfaces configured with the ip igmp mroute-proxy command (see Step 7) based on the IGMP query interval. Where there is a match, one IGMP report is created and received on this interface.</p> <p>Note The ip igmp proxy-service command is intended to be used with the ip igmp helper-address command.</p> |

| | Command or Action | Purpose |
|----------------|--|---|
| Step 9 | end Example: Device(config-if)# end | Ends the current configuration session and returns to privileged EXEC mode. |
| Step 10 | show ip igmp interface Example: Device# show ip igmp interface | (Optional) Displays multicast-related information about an interface. |

Configuring the Downstream Device for IGMP Proxy for Multiple Upstream Interfaces

Perform this task to configure the downstream device for IGMP Proxy for multiple upstream interfaces.

(Referring to the [figure](#) earlier, all the steps are configured on the *Router B* interface facing the upstream device)

Procedure

| | Command or Action | Purpose |
|---------------|---|---|
| Step 1 | enable Example: Device > enable | Enables privileged EXEC mode. Enter your password if prompted. |
| Step 2 | configure terminal Example: Device# configure terminal | Enters global configuration mode. |
| Step 3 | class-map type multicast-flows name Example: Device(config-if)# class-map type multicast-flows proxymap | Configures the interface with a class-map where the uplink interfaces for different multicast groups are defined. The range of the multicast groups is from group 225.0.0.1 to 225.0.0.10. |
| Step 4 | interface type number Example: Device(config)# interface ethernet 0/0 | Enters interface configuration mode. For the <i>type</i> and <i>number</i> arguments, specify the interface that is facing the host. |
| Step 5 | ip igmp upstream-proxy class-map-name Example: Device(config-if)# ip igmp upstream-proxy proxymap | Enables the interface with an IGMP proxy. The IGMP proxy joins for those groups in the class map are sent if the following conditions are met: <ul style="list-style-type: none"> • (*,G) or (S,G) mroute exists for the group in the same mvrfl as the interface. • (*,G) or (S,G) mroute has a NON-NULl OIF list. |

| | Command or Action | Purpose |
|---------------|--|--|
| Step 6 | ip igmp iif-starg Example: Device(config-if)# ip igmp iif-starg | Alters the RPF interface of the mroutes of those groups specified in the class map to Ethernet 0/0. |
| Step 7 | ip igmp proxy-report-interval time Example: Device(config-if)# ip igmp proxy-report-interval 130 | Configures the interval (in seconds) at which the proxy reports will be sent. The default value is 60 seconds. |
| Step 8 | end Example: Device(config-if)# end | Ends the current configuration session and returns to privileged EXEC mode. |
| Step 9 | show ip igmp interface Example: Device# show ip igmp interface | (Optional) Displays multicast-related information about an interface. |

Configuration Examples for IGMP Proxy

Example: Configuring the Upstream UDL Device for IGMP UDLR

The following example shows how to configure the upstream UDL device for IGMP UDLR:

```
interface gigabitethernet 0/0/0
ip address 10.1.1.1 255.255.255.0
ip pim sparse-mode
!
interface gigabitethernet 1/0/0
ip address 10.2.1.1 255.255.255.0
ip pim sparse-mode
ip igmp unidirectional-link
!
interface gigabitethernet 2/0/0
ip address 10.3.1.1 255.255.255.0
```

Example: Configuring the Downstream UDL Device for IGMP UDLR with IGMP Proxy Support

The following example shows how to configure the downstream UDL device for IGMP UDLR with IGMP proxy support:

```
ip pim rp-address 10.5.1.1 5
access-list 5 permit 239.0.0.0 0.255.255.255
!
interface loopback 0
ip address 10.7.1.1 255.255.255.0
```

```

ip pim sparse-mode
ip igmp helper-address udl ethernet 0
ip igmp proxy-service
!
interface gigabitethernet 0/0/0
ip address 10.2.1.2 255.255.255.0
ip pim sparse-mode
ip igmp unidirectional-link
!
interface gigabitethernet 1/0/0
ip address 10.5.1.1 255.255.255.0
ip pim sparse-mode
ip igmp mroute-proxy loopback 0
!
interface gigabitethernet 2/0/0
ip address 10.6.1.1 255.255.255.0

```

Example: Configuring the Downstream Device for IGMP Proxy Join without UDLR

The following example shows how to configure the downstream device for IGMP proxy without UDLR:

```

interface Loopback0
ip address 2.2.2.2 255.255.0.0
ip pim sparse-dense-mode
ip igmp helper-address 99.99.99.1
ip igmp proxy-service
ip ospf 1 area 0
!

```

Example: Configuring the Downstream Device for IGMP Proxy for Multiple Upstream Interfaces

The following example shows how to configure the downstream device for IGMP proxy for multiple upstream interfaces.

```

interface gigabitethernet0/0
ip address 99.99.99.1 255.255.255.0
ip pim passive
ip igmp upstream-proxy 12
ip igmp iif-starg
ip igmp proxy-report-interval 100
end

class-map type multicast-flows 12
group 229.0.0.1
group 228.0.0.1 to 228.0.0.10

```

Additional References for IGMP Proxy

The following sections provide references related to customizing IGMP.

Related Documents

| Related Topic | Document Title |
|--|---|
| For complete syntax and usage information for the commands used in this chapter. | See the IP Multicast Routing Commands section of the <i>Command Reference (Catalyst 9200 Series Switches)</i> |

Standards and RFCs

| Standard/RFC | Title |
|--------------|--|
| RFC 1112 | <i>Host extensions for IP multicasting</i> |
| RFC 2236 | <i>Internet Group Management Protocol, Version 2</i> |
| RFC 3376 | <i>Internet Group Management Protocol, Version 3</i> |

Feature History for IGMP Proxy

This table provides release and related information for the features explained in this module.

These features are available in all the releases subsequent to the one they were introduced in, unless noted otherwise.

| Release | Feature | Feature Information |
|--------------------------------|---|--|
| Cisco IOS XE Fuji 16.9.2 | IGMP Proxy | An IGMP proxy enables hosts in a unidirectional link routing (UDLR) environment that are not directly connected to a downstream router to join a multicast group sourced from an upstream network. |
| Cisco IOS XE Gibraltar 16.12.1 | IGMP Proxy without UDLR | The IGMP proxy enables hosts to receive traffic from an upstream network, without using a UDLR link. |
| Cisco IOS XE Amsterdam 17.1.1 | IGMP Proxy for multiple upstream interfaces. | IGMP proxy enables the user to receive traffic from multiple upstream devices. |
| Cisco IOS XE Cupertino 17.7.1 | IGMP Proxy IGMP Proxy without UDLR IGMP Proxy for multiple upstream interfaces | These features were implemented on supervisor modules C9400X-SUP-2 and C9400X-SUP-2XL, which were introduced in this release. |
| Cisco IOS XE Cupertino 17.9.1 | IGMP Proxy IGMP Proxy without UDLR IGMP Proxy for multiple upstream interfaces. | This feature was implemented on C9200CX-12P-2X2G, C9200CX-8P-2X2G, and C9200CX-12T-2X2G models of the Cisco Catalyst 9200CX Series Switches, which were introduced in this release. |

Use the Cisco Feature Navigator to find information about platform and software image support. To access Cisco Feature Navigator, go to <http://www.cisco.com/go/cfn>.



CHAPTER 6

Constraining IP Multicast in Switched Ethernet

- [Prerequisites for Constraining IP Multicast in a Switched Ethernet Network, on page 111](#)
- [Information About IP Multicast in a Switched Ethernet Network, on page 111](#)
- [How to Constrain Multicast in a Switched Ethernet Network, on page 113](#)
- [Configuration Examples for Constraining IP Multicast in a Switched Ethernet Network, on page 115](#)
- [Additional References for Constraining IP Multicast in a Switched Ethernet Network, on page 116](#)
- [Feature History for Constraining IP Multicast in Switched Ethernet, on page 116](#)

Prerequisites for Constraining IP Multicast in a Switched Ethernet Network

Before using the tasks in this module, you should be familiar with the concepts described in the [IP Multicast Routing Technology Overview, on page 1](#) module.

Information About IP Multicast in a Switched Ethernet Network

IP Multicast Traffic and Layer 2 Switches

The default behavior for a Layer 2 switch is to forward all multicast traffic to every port that belongs to the destination LAN on the switch. This behavior reduces the efficiency of the switch, whose purpose is to limit traffic to the ports that need to receive the data. This behavior requires a constraining mechanism to reduce unnecessary multicast traffic, which improves switch performance.

Cisco Group Management Protocol (CGMP), Router Group Management Protocol (RGMP), and IGMP snooping efficiently constrain IP multicast in a Layer 2 switching environment.

- CGMP and IGMP snooping are used on subnets that include end users or receiver clients.
- RGMP is used on routed segments that contain only routers, such as in a collapsed backbone.
- RGMP and CGMP cannot interoperate. However, Internet Group Management Protocol (IGMP) can interoperate with CGMP and RGMP snooping.

CGMP on Catalyst Switches for IP Multicast

CGMP is a Cisco-developed protocol used on device connected to Catalyst switches to perform tasks similar to those performed by IGMP. CGMP is necessary for those Catalyst switches that do not distinguish between IP multicast data packets and IGMP report messages, both of which are addressed to the same group address at the MAC level. The switch can distinguish IGMP packets, but would need to use software on the switch, greatly impacting its performance.

You must configure CGMP on the multicast device and the Layer 2 switches. The result is that, with CGMP, IP multicast traffic is delivered only to those Catalyst switch ports that are attached to interested receivers. All other ports that have not explicitly requested the traffic will not receive it unless these ports are connected to a multicast router. Multicast router ports must receive every IP multicast data packet.

Using CGMP, when a host joins a multicast group, it multicasts an unsolicited IGMP membership report message to the target group. The IGMP report is passed through the switch to the router for normal IGMP processing. The router (which must have CGMP enabled on this interface) receives the IGMP report and processes it as it normally would, but also creates a CGMP Join message and sends it to the switch. The Join message includes the MAC address of the end station and the MAC address of the group it has joined.

The switch receives this CGMP Join message and then adds the port to its content-addressable memory (CAM) table for that multicast group. All subsequent traffic directed to this multicast group is then forwarded out the port for that host.

The Layer 2 switches are designed so that several destination MAC addresses could be assigned to a single physical port. This design allows switches to be connected in a hierarchy and also allows many multicast destination addresses to be forwarded out a single port.

The device port also is added to the entry for the multicast group. Multicast device must listen to all multicast traffic for every group because IGMP control messages are also sent as multicast traffic. The rest of the multicast traffic is forwarded using the CAM table with the new entries created by CGMP.

IGMP Snooping

IGMP snooping is an IP multicast constraining mechanism that runs on a Layer 2 LAN switch. IGMP snooping requires the LAN switch to examine, or “snoop,” some Layer 3 information (IGMP Join/Leave messages) in the IGMP packets sent between the hosts and the router. When the switch receives the IGMP host report from a host for a particular multicast group, the switch adds the port number of the host to the associated multicast table entry. When the switch hears the IGMP Leave group message from a host, the switch removes the table entry of the host.

Because IGMP control messages are sent as multicast packets, they are indistinguishable from multicast data at Layer 2. A switch running IGMP snooping must examine every multicast data packet to determine if it contains any pertinent IGMP control information. IGMP snooping implemented on a low-end switch with a slow CPU could have a severe performance impact when data is sent at high rates. The solution is to implement IGMP snooping on high-end switches with special application-specific integrated circuits (ASICs) that can perform the IGMP checks in hardware. CGMP is a better option for low-end switches without special hardware.

Router-Port Group Management Protocol (RGMP)

CGMP and IGMP snooping are IP multicast constraining mechanisms designed to work on routed network segments that have active receivers. They both depend on IGMP control messages that are sent between the hosts and the routers to determine which switch ports are connected to interested receivers.

Switched Ethernet backbone network segments typically consist of several routers connected to a switch without any hosts on that segment. Because routers do not generate IGMP host reports, CGMP and IGMP snooping will not be able to constrain the multicast traffic, which will be flooded to every port on the VLAN. Routers instead generate Protocol Independent Multicast (PIM) messages to Join and Prune multicast traffic flows at a Layer 3 level.

Router-Port Group Management Protocol (RGMP) is an IP multicast constraining mechanism for router-only network segments. RGMP must be enabled on the routers and on the Layer 2 switches. A multicast router indicates that it is interested in receiving a data flow by sending an RGMP Join message for a particular group. The switch then adds the appropriate port to its forwarding table for that multicast group--similar to the way it handles a CGMP Join message. IP multicast data flows will be forwarded only to the interested router ports. When the router no longer is interested in that data flow, it sends an RGMP Leave message and the switch removes the forwarding entry.

If there are any routers that are not RGMP-enabled, they will continue to receive all multicast data.

How to Constrain Multicast in a Switched Ethernet Network

Configuring Switches for IP Multicast

If you have switching in your multicast network, consult the documentation for the switch you are working with for information about how to configure IP multicast.

Configuring IGMP Snooping

No configuration is required on the router. Consult the documentation for the switch you are working with to determine how to enable IGMP snooping and follow the provided instructions.

Enabling CGMP

CGMP is a protocol used on devices connected to Catalyst switches to perform tasks similar to those performed by IGMP. CGMP is necessary because the Catalyst switch cannot distinguish between IP multicast data packets and IGMP report messages, which are both at the MAC level and are addressed to the same group address.



Note

- CGMP should be enabled only on 802 or ATM media, or LAN emulation (LANE) over ATM.
- CGMP should be enabled only on devices connected to Catalyst switches.

Procedure

| | Command or Action | Purpose |
|--------|------------------------|---|
| Step 1 | enable Example: | Enables privileged EXEC mode. • Enter your password if prompted. |

| | Command or Action | Purpose |
|---------------|---|--|
| | Device> enable | |
| Step 2 | configure terminal Example: Device# configure terminal | Enters global configuration mode. |
| Step 3 | interface <i>type number</i> Example: Device(config)# interface ethernet 1 | Selects an interface that is connected to hosts on which IGMPv3 can be enabled. |
| Step 4 | ip cgmp [proxy router-only] Example: Device(config-if)# ip cgmp proxy | Enables CGMP on an interface of a device connected to a Cisco Catalyst 5000 family switch. <ul style="list-style-type: none"> The proxy keyword enables the CGMP proxy function. When enabled, any device that is not CGMP-capable will be advertised by the proxy router. The proxy router advertises the existence of other non-CGMP-capable devices by sending a CGMP Join message with the MAC address of the non-CGMP-capable device and group address of 0000.0000.0000. |
| Step 5 | end Example: Device(config-if)# end | Ends the current configuration session and returns to EXEC mode. |
| Step 6 | clear ip cgmp [<i>interface-type interface-number</i>] Example: Device# clear ip cgmp | (Optional) Clears all group entries from the caches of Catalyst switches. |

Configuring IP Multicast in a Layer 2 Switched Ethernet Network

Perform this task to configure IP multicast in a Layer 2 Switched Ethernet network using RGMP.

Procedure

| | Command or Action | Purpose |
|---------------|----------------------------------|--|
| Step 1 | enable Example: | Enables privileged EXEC mode. <ul style="list-style-type: none"> Enter your password if prompted. |

| | Command or Action | Purpose |
|---------------|--|---|
| | Device> enable | |
| Step 2 | configure terminal Example: Device# configure terminal | Enters global configuration mode. |
| Step 3 | interface <i>type number</i> Example: Device(config)# interface ethernet 1 | Selects an interface that is connected to hosts. |
| Step 4 | ip rgmp Example: Device(config-if)# ip rgmp | Enables RGMP on Ethernet, Fast Ethernet, and Gigabit Ethernet interfaces. |
| Step 5 | end Example: Device(config-if)# end | Ends the current configuration session and returns to EXEC mode. |
| Step 6 | debug ip rgmp Example: Device# debug ip rgmp | (Optional) Logs debug messages sent by an RGMP-enabled device. |
| Step 7 | show ip igmp interface Example: Device# show ip igmp interface | (Optional) Displays multicast-related information about an interface. |

Configuration Examples for Constraining IP Multicast in a Switched Ethernet Network

RGMP Configuration Example

The following example shows how to configure RGMP on a router:

```
ip multicast-routing
ip pim sparse-mode
interface ethernet 0
 ip rgmp
```

Additional References for Constraining IP Multicast in a Switched Ethernet Network

Related Documents

| Related Topic | Document Title |
|--|---|
| For complete syntax and usage information for the commands used in this chapter. | See the IP Multicast Routing Commands section of the <i>Command Reference (Catalyst 9200 Series Switches)</i> |

MIBs

| MIB | MIBs Link |
|---|---|
| No new or modified MIBs are supported by these features, and support for existing MIBs has not been modified by these features. | To locate and download MIBs for selected platforms, Cisco IOS XE releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs |

Technical Assistance

| Description | Link |
|---|---|
| The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password. | http://www.cisco.com/cisco/web/support/index.html |

Feature History for Constraining IP Multicast in Switched Ethernet

This table provides release and related information for the features explained in this module.

These features are available in all the releases subsequent to the one they were introduced in, unless noted otherwise.

| Release | Feature | Feature Information |
|--------------------------|--|---|
| Cisco IOS XE Fuji 16.9.2 | Constraining IP Multicast in Switched Ethernet | IP multicast in switched ethernet provides a constraining mechanism to reduce unnecessary multicast traffic, which improves switch performance. |

| Release | Feature | Feature Information |
|-------------------------------|--|---|
| Cisco IOS XE Cupertino 17.7.1 | Constraining IP Multicast in Switched Ethernet | This feature was implemented on supervisor modules C9400X-SUP-2 and C9400X-SUP-2XL, which were introduced in this release. |
| Cisco IOS XE Cupertino 17.9.1 | Constraining IP Multicast in Switched Ethernet | This feature was implemented on C9200CX-12P-2X2G, C9200CX-8P-2X2G, and C9200CX-12T-2X2G models of the Cisco Catalyst 9200CX Series Switches, which were introduced in this release. |

Use the Cisco Feature Navigator to find information about platform and software image support. To access Cisco Feature Navigator, go to <http://www.cisco.com/go/cfn>.



CHAPTER 7

Configuring Protocol Independent Multicast (PIM)

- [Prerequisites for PIM, on page 119](#)
- [Restrictions for PIM, on page 120](#)
- [Information about PIM, on page 122](#)
- [How to Configure PIM, on page 136](#)
- [Verifying PIM Operations, on page 160](#)
- [Monitoring and Troubleshooting PIM, on page 169](#)
- [Configuration Examples for PIM, on page 171](#)
- [Feature History for PIM, on page 173](#)

Prerequisites for PIM

Before you begin the PIM configuration process, decide which PIM mode to use. This is based on the applications you intend to support on your network. Use the following guidelines:

- In general, if the application is one-to-many or many-to-many in nature, then PIM-SM can be used successfully.
- For optimal one-to-many application performance, SSM is appropriate but requires IGMP version 3 support.

Before you configure PIM stub routing, check that you have met these conditions:

- You must have IP multicast routing configured on both the stub router and the central router. You must also have PIM mode configured on the uplink interface of the stub router.
- You must also configure either Enhanced Interior Gateway Routing Protocol (EIGRP) stub routing or Open Shortest Path First (OSPF) stub routing on the device.
- The PIM stub router does not route the transit traffic between the distribution routers. Unicast (EIGRP) stub routing enforces this behavior. You must configure unicast stub routing to assist the PIM stub router behavior.

Restrictions for PIM

The following are the restrictions for configuring PIM:

- Use ACLs to designate a specified port only as a multicast host port and not as a multicast router port. Multicast router control-packets received on this port are dropped.
- PIM nonbroadcast multiaccess (NBMA) mode is not supported on an ethernet interface.
- Hot Standby Router Protocol-aware (HSRP-aware) PIM is not supported.

PIMv1 and PIMv2 Interoperability

To avoid misconfiguring multicast routing on your device, review the information in this section.

The Cisco PIMv2 implementation provides interoperability and transition between Version 1 and Version 2, although there might be some minor problems.

You can upgrade to PIMv2 incrementally. PIM Versions 1 and 2 can be configured on different routers and multilayer switches within one network. Internally, all routers and multilayer switches on a shared media network must run the same PIM version. Therefore, if a PIMv2 device detects a PIMv1 device, the Version 2 device downgrades itself to Version 1 until all Version 1 devices have been shut down or upgraded.

PIMv2 uses the BSR to discover and announce RP-set information for each group prefix to all the routers and multilayer switches in a PIM domain. PIMv1, together with the Auto-RP feature, can perform the same tasks as the PIMv2 BSR. However, Auto-RP is a standalone protocol, separate from PIMv1, and is a proprietary Cisco protocol. PIMv2 is a standards track protocol in the IETF.



Note We recommend that you use PIMv2. The BSR function interoperates with Auto-RP on Cisco routers and multilayer switches.

When PIMv2 devices interoperate with PIMv1 devices, Auto-RP should have already been deployed. A PIMv2 BSR that is also an Auto-RP mapping agent automatically advertises the RP elected by Auto-RP. That is, Auto-RP sets its single RP on every router or multilayer switch in the group. Not all routers and switches in the domain use the PIMv2 hash function to select multiple RPs.

Sparse-mode groups in a mixed PIMv1 and PIMv2 region are possible because the Auto-RP feature in PIMv1 interoperates with the PIMv2 RP feature. Although all PIMv2 devices can also use PIMv1, we recommend that the RPs be upgraded to PIMv2. To ease the transition to PIMv2, we recommend:

- Using Auto-RP throughout the region.

If Auto-RP is not already configured in the PIMv1 regions, configure Auto-RP.

Restrictions for Configuring PIM Stub Routing

- Only directly connected multicast (IGMP) receivers and sources are allowed in the Layer 2 access domains. The PIM protocol is not supported in access domains.

- In a network using PIM stub routing, the only allowable route for IP traffic to the user is through a device that is configured with PIM stub routing.
- The redundant PIM stub router topology is not supported. Only the nonredundant access router topology is supported by the PIM stub feature.

Restrictions for Configuring Auto-RP and BSR

Take into consideration your network configuration, and the following restrictions when configuring Auto-RP and BSR:

Restrictions for Configuring Auto-RP

The following are restrictions for configuring Auto-RP (if used in your network configuration):

- If routed interfaces are configured in sparse mode, Auto-RP can still be used if all devices are configured with a manual RP address for the Auto-RP groups.
- If routed interfaces are configured in sparse mode and you enter the **ip pim autorp listener** global configuration command, Auto-RP can still be used even if all devices are not configured with a manual RP address for the Auto-RP groups.

Restrictions for Configuring BSR

The following are the restrictions for configuring BSR (if used in your network configuration):

- Configure the candidate BSRs as the RP-mapping agents for Auto-RP.
- For group prefixes advertised through Auto-RP, the PIMv2 BSR mechanism should not advertise a subrange of these group prefixes served by a different set of RPs. In a mixed PIMv1 and PIMv2 domain, have backup RPs serve the same group prefixes. This prevents the PIMv2 DRs from selecting a different RP from those PIMv1 DRs, due to the longest match lookup in the RP-mapping database.

Restrictions and Guidelines for Configuring Auto-RP and BSR

The following are restrictions for configuring Auto-RP and BSR (if used in your network configuration):

- If your network is all Cisco routers and multilayer switches, you can use either Auto-RP or BSR.
- If you have non-Cisco routers in your network, you must use BSR.
- If you have Cisco PIMv1 and PIMv2 routers and multilayer switches and non-Cisco routers, you must use both Auto-RP and BSR. If your network includes routers from other vendors, configure the Auto-RP mapping agent and the BSR on a Cisco PIMv2 device. Ensure that no PIMv1 device is located in the path a between the BSR and a non-Cisco PIMv2 device.



Note There are two approaches to using PIMv2. You can use Version 2 exclusively in your network or migrate to Version 2 by employing a mixed PIM version environment.

- Because bootstrap messages are sent hop-by-hop, a PIMv1 device prevents these messages from reaching all routers and multilayer switches in your network. Therefore, if your network has a PIMv1 device in it and only Cisco routers and multilayer switches, it is best to use Auto-RP.
- If you have a network that includes non-Cisco routers, configure the Auto-RP mapping agent and the BSR on a Cisco PIMv2 router or multilayer switch. Ensure that no PIMv1 device is on the path between the BSR and a non-Cisco PIMv2 router.
- If you have non-Cisco PIMv2 routers that need to interoperate with Cisco PIMv1 routers and multilayer switches, both Auto-RP and a BSR are required. We recommend that a Cisco PIMv2 device be both the Auto-RP mapping agent and the BSR.

Restrictions for Auto-RP Enhancement

The simultaneous deployment of Auto-RP and bootstrap router (BSR) is not supported.

Information about PIM

Protocol Independent Multicast Overview

The Protocol Independent Multicast (PIM) protocol maintains the current IP multicast service mode of receiver-initiated membership. PIM is not dependent on a specific unicast routing protocol; it is IP routing protocol independent and can leverage whichever unicast routing protocols are used to populate the unicast routing table, including Enhanced Interior Gateway Routing Protocol (EIGRP), Open Shortest Path First (OSPF), and static routes. PIM uses unicast routing information to perform the multicast forwarding function.

Although PIM is called a multicast routing protocol, it actually uses the unicast routing table to perform the reverse path forwarding (RPF) check function instead of building up a completely independent multicast routing table. Unlike other routing protocols, PIM does not send and receive routing updates between routers.

PIM is defined in RFC 4601, Protocol Independent Multicast - Sparse Mode (PIM-SM)

PIM Versions

PIMv2 includes these improvements over PIMv1:

- A single, active rendezvous point (RP) exists per multicast group, with multiple backup RPs. This single RP compares to multiple active RPs for the same group in PIMv1.
- A bootstrap router (BSR) provides a fault-tolerant, automated RP discovery and distribution function that enables routers and multilayer switches to dynamically learn the group-to-RP mappings.
- PIM join and prune messages have more flexible encoding for multiple address families.
- A more flexible hello packet format replaces the query packet to encode current and future capability options.
- Register messages sent to an RP specify whether they are sent by a border router or a designated router.
- PIM packets are no longer inside IGMP packets; they are standalone packets.

Multicast Source Discovery Protocol (MSDP)

Multicast Source Discovery Protocol (MSDP) is used for inter-domain source discovery when PIM SM is used. Each PIM administrative domain has its own RP. In order for the RP in one domain to signal new sources to the RP in the other domain, MSDP is used.

When RP in a domain receives a PIM register message for a new source, with MSDP configured it sends a new source-active (SA) message to all its MSDP peers in other domains. Each intermediate MSDP peer floods this SA message away from the originating RP. The MSDP peers install this SA message in their MSDP sa-cache. If the RPs in other domains have any join requests for the group in the SA message (indicated by the presence of a (*,G) entry with non empty outgoing interface list), the domain is interested in the group, and the RP triggers an (S,G) join toward the source.

PIM Sparse Mode

PIM sparse mode (PIM-SM) uses a pull model to deliver multicast traffic. Only network segments with active receivers that have explicitly requested the data will receive the traffic.

Sparse mode interfaces are added to the multicast routing table only when periodic Join messages are received from downstream routers, or when a directly connected member is on the interface. When forwarding from a LAN, sparse mode operation occurs if an RP is known for the group. If so, the packets are encapsulated and sent toward the RP. If the multicast traffic from a specific source is sufficient, the first hop router of the receiver may send Join messages toward the source to build a source-based distribution tree.

PIM-SM distributes information about active sources by forwarding data packets on the shared tree. Because PIM-SM uses shared trees (at least, initially), it requires the use of a rendezvous point (RP). The RP must be administratively configured in the network. See the [Rendezvous Points, on page 125](#) section for more information.

In sparse mode, a router assumes that other routers do not want to forward multicast packets for a group, unless there is an explicit request for the traffic. When hosts join a multicast group, the directly connected routers send PIM Join messages toward the RP. The RP keeps track of multicast groups. Hosts that send multicast packets are registered with the RP by the first hop router of that host. The RP then sends Join messages toward the source. At this point, packets are forwarded on a shared distribution tree. If the multicast traffic from a specific source is sufficient, the first hop router of the host may send Join messages toward the source to build a source-based distribution tree.

Sources register with the RP and then data is forwarded down the shared tree to the receivers. The edge routers learn about a particular source when they receive data packets on the shared tree from that source through the RP. The edge router then sends PIM (S,G) Join messages toward that source. Each router along the reverse path compares the unicast routing metric of the RP address to the metric of the source address. If the metric for the source address is better, it will forward a PIM (S,G) Join message toward the source. If the metric for the RP is the same or better, then the PIM (S,G) Join message will be sent in the same direction as the RP. In this case, the shared tree and the source tree would be considered congruent.

If the shared tree is not an optimal path between the source and the receiver, the routers dynamically create a source tree and stop traffic from flowing down the shared tree. This behavior is the default behavior in software. Network administrators can force traffic to stay on the shared tree by using the **ip pim spt-threshold infinity** command.

PIM-SM scales well to a network of any size, including those with WAN links. The explicit join mechanism prevents unwanted traffic from flooding the WAN links.

PIM Stub Routing

The PIM stub routing feature, available in all of the device software images, reduces resource usage by moving routed traffic closer to the end user.

The PIM stub routing feature supports multicast routing between the distribution layer and the access layer. It supports two types of PIM interfaces, uplink PIM interfaces, and PIM passive interfaces. A routed interface configured with the PIM passive mode does not pass or forward PIM control traffic, it only passes and forwards IGMP traffic.

In a network using PIM stub routing, the only allowable route for IP traffic to the user is through a device that is configured with PIM stub routing. PIM passive interfaces are connected to Layer 2 access domains, such as VLANs, or to interfaces that are connected to other Layer 2 devices. Only directly connected multicast (IGMP) receivers and sources are allowed in the Layer 2 access domains. The PIM passive interfaces do not send or process any received PIM control packets.

When using PIM stub routing, you should configure the distribution and remote routers to use IP multicast routing and configure only the device as a PIM stub router. The device does not route transit traffic between distribution routers. You also need to configure a routed uplink port on the device. The device uplink port cannot be used with SVIs. If you need PIM for an SVI uplink port, you should upgrade to the Network Advantage license.

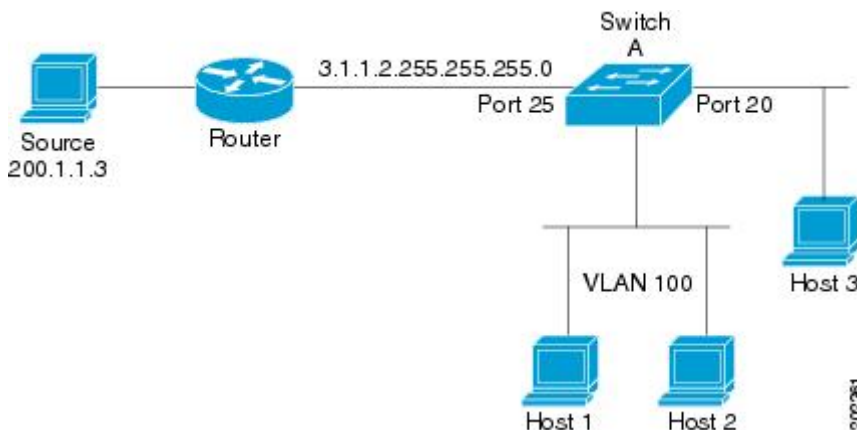


Note You must also configure EIGRP stub routing when configuring PIM stub routing on the device

The redundant PIM stub router topology is not supported. The redundant topology exists when there is more than one PIM router forwarding multicast traffic to a single access domain. PIM messages are blocked, and the PIM asset and designated router election mechanisms are not supported on the PIM passive interfaces. Only the nonredundant access router topology is supported by the PIM stub feature. By using a nonredundant topology, the PIM passive interface assumes that it is the only interface and designated router on that access domain.

Figure 10: PIM Stub Router Configuration

In the following figure, the Device A routed uplink port 25 is connected to the router and PIM stub routing is enabled on the VLAN 100 interfaces and on Host 3. This configuration allows the directly connected hosts to receive traffic from multicast source 200.1.1.3.



Rendezvous Points

A rendezvous point (RP) is a role that a device performs when operating in Protocol Independent Multicast (PIM) Sparse Mode (SM). An RP is required only in networks running PIM SM. In the PIM-SM model, only network segments with active receivers that have explicitly requested multicast data will be forwarded the traffic.

An RP acts as the meeting place for sources and receivers of multicast data. In a PIM-SM network, sources must send their traffic to the RP. This traffic is then forwarded to receivers down a shared distribution tree. By default, when the first hop device of the receiver learns about the source, it will send a Join message directly to the source, creating a source-based distribution tree from the source to the receiver. This source tree does not include the RP unless the RP is located within the shortest path between the source and receiver.

In most cases, the placement of the RP in the network is not a complex decision. By default, the RP is needed only to start new sessions with sources and receivers. Consequently, the RP experiences little overhead from traffic flow or processing. In PIM version 2, the RP performs less processing than in PIM version 1 because sources must only periodically register with the RP to create state.

Auto-RP

In the first version of PIM-SM, all leaf routers (routers directly connected to sources or receivers) were required to be manually configured with the IP address of the RP. This type of configuration is also known as static RP configuration. Configuring static RPs is relatively easy in a small network, but it can be laborious in a large, complex network.

Following the introduction of PIM-SM version 1, Cisco implemented a version of PIM-SM with the Auto-RP feature. Auto-RP automates the distribution of group-to-RP mappings in a PIM network. Auto-RP has the following benefits:

- Configuring the use of multiple RPs within a network to serve different groups is easy.
- Auto-RP allows load splitting among different RPs and arrangement of RPs according to the location of group participants.
- Auto-RP avoids inconsistent, manual RP configurations that can cause connectivity problems.

Multiple RPs can be used to serve different group ranges or serve as backups to each other. For Auto-RP to work, a router must be designated as an RP-mapping agent, which receives the RP-announcement messages from the RPs and arbitrates conflicts. The RP-mapping agent then sends the consistent group-to-RP mappings to all other routers. Thus, all routers automatically discover which RP to use for the groups they support.



Note If router interfaces are configured in sparse mode, Auto-RP can still be used if all routers are configured with a static RP address for the Auto-RP groups.

To make Auto-RP work, a router must be designated as an RP mapping agent, which receives the RP announcement messages from the RPs and arbitrates conflicts. Thus, all routers automatically discover which RP to use for the groups they support. The Internet Assigned Numbers Authority (IANA) has assigned two group addresses, 224.0.1.39 and 224.0.1.40, for Auto-RP. One advantage of Auto-RP is that any change to the RP designation must be configured only on the routers that are RPs and not on the leaf routers. Another advantage of Auto-RP is that it offers the ability to scope the RP address within a domain. Scoping can be achieved by defining the time-to-live (TTL) value allowed for the Auto-RP advertisements.

Each method for configuring an RP has its own strengths, weaknesses, and level of complexity. In conventional IP multicast network scenarios, we recommend using Auto-RP to configure RPs because it is easy to configure, well-tested, and stable. The alternative ways to configure an RP are static RP, Auto-RP, and bootstrap router.

The Role of Auto-RP in a PIM Network

Auto-RP automates the distribution of group-to-rendezvous point (RP) mappings in a PIM network. To make Auto-RP work, a device must be designated as an RP mapping agent, which receives the RP announcement messages from the RPs and arbitrates conflicts.

Thus, all routers automatically discover which RP to use for the groups they support. The Internet Assigned Numbers Authority (IANA) has assigned two group addresses, 224.0.1.39 and 224.0.1.40, for Auto-RP.

The mapping agent receives announcements of intention to become the RP from Candidate-RPs. The mapping agent then announces the winner of the RP election. This announcement is made independently of the decisions by the other mapping agents.

Multicast Boundaries

Administratively-scoped boundaries can be used to limit the forwarding of multicast traffic outside of a domain or subdomain. This approach uses a special range of multicast addresses, called administratively-scoped addresses, as the boundary mechanism. If you configure an administratively-scoped boundary on a routed interface, multicast traffic whose multicast group addresses fall in this range cannot enter or exit this interface, which provides a firewall for multicast traffic in this address range.

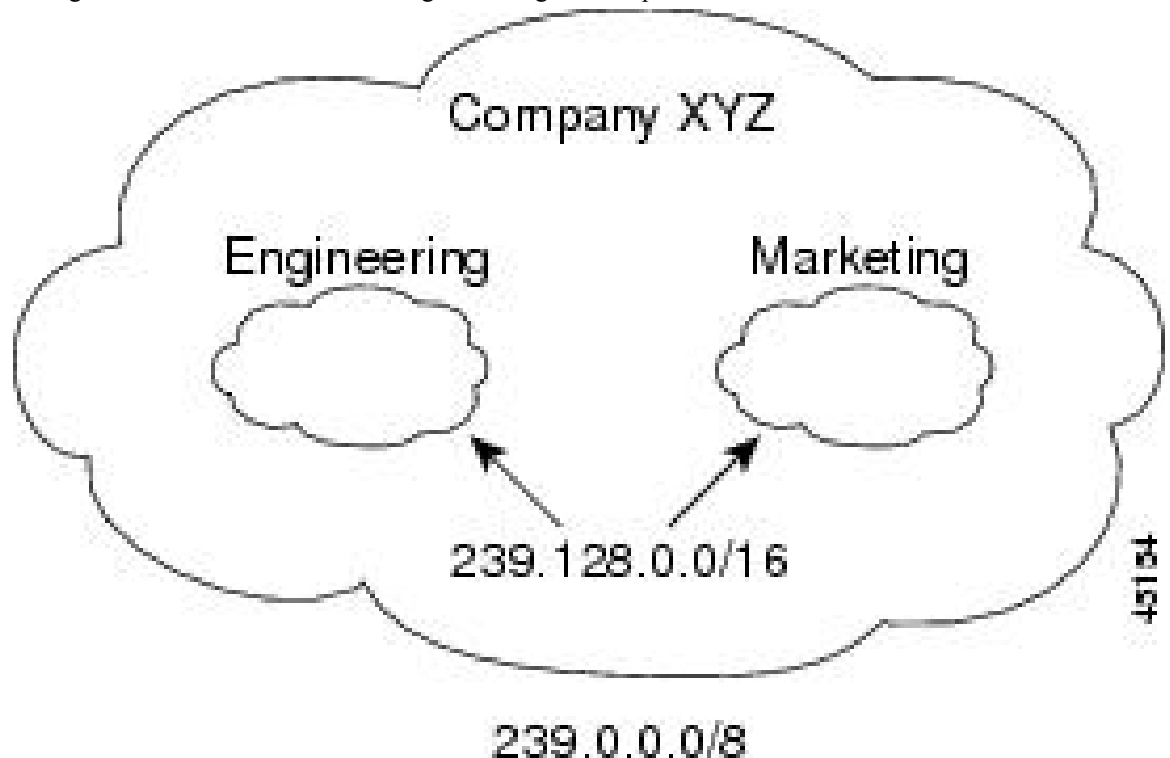


Note Multicast boundaries and TTL thresholds control the scoping of multicast domains; however, TTL thresholds are not supported by the device. You should use multicast boundaries instead of TTL thresholds to limit the forwarding of multicast traffic outside of a domain or a subdomain.

Figure 11: Administratively-Scoped Boundaries

The following figure shows that Company XYZ has an administratively-scoped boundary set for the multicast address range 239.0.0.0/8 on all routed interfaces at the perimeter of its network. This boundary prevents any multicast traffic in the range 239.0.0.0 through 239.255.255.255 from entering or leaving the network. Similarly, the engineering and marketing departments have an administratively-scoped boundary of 239.128.0.0/16 around the perimeter of their networks. This boundary prevents multicast traffic in the range of 239.128.0.0

through 239.128.255.255 from entering or leaving their respective networks.



You can define an administratively-scoped boundary on a routed interface for multicast group addresses. A standard access list defines the range of addresses affected. When a boundary is defined, no multicast data packets are allowed to flow across the boundary from either direction. The boundary allows the same multicast group address to be reused in different administrative domains.

The IANA has designated the multicast address range 239.0.0.0 to 239.255.255.255 as the administratively-scoped addresses. This range of addresses can then be reused in domains administered by different organizations. The addresses would be considered local, not globally unique.

You can configure the **filter-autorp** keyword to examine and filter Auto-RP discovery and announcement messages at the administratively scoped boundary. Any Auto-RP group range announcements from the Auto-RP packets that are denied by the boundary access control list (ACL) are removed. An Auto-RP group range announcement is permitted and passed by the boundary only if all addresses in the Auto-RP group range are permitted by the boundary ACL. If any address is not permitted, the entire group range is filtered and removed from the Auto-RP message before the Auto-RP message is forwarded.

Sparse-Dense Mode for Auto-RP

A prerequisite of Auto-RP is that all interfaces must be configured in sparse-dense mode using the **ip pim sparse-dense-mode** interface configuration command. An interface configured in sparse-dense mode is treated in either sparse mode or dense mode of operation, depending on which mode the multicast group operates. If a multicast group has a known RP, the interface is treated in sparse mode. If a group has no known RP, by default the interface is treated in dense mode and data will be flooded over this interface. (You can prevent dense-mode fallback; see the module “Configuring Basic IP Multicast.”)

To successfully implement Auto-RP and prevent any groups other than 224.0.1.39 and 224.0.1.40 from operating in dense mode, we recommend configuring a “sink RP” (also known as “RP of last resort”). A sink

RP is a statically configured RP that may or may not actually exist in the network. Configuring a sink RP does not interfere with Auto-RP operation because, by default, Auto-RP messages supersede static RP configurations. We recommend configuring a sink RP for all possible multicast groups in your network, because it is possible for an unknown or unexpected source to become active. If no RP is configured to limit source registration, the group may revert to dense mode operation and be flooded with data.

Auto RP Benefits

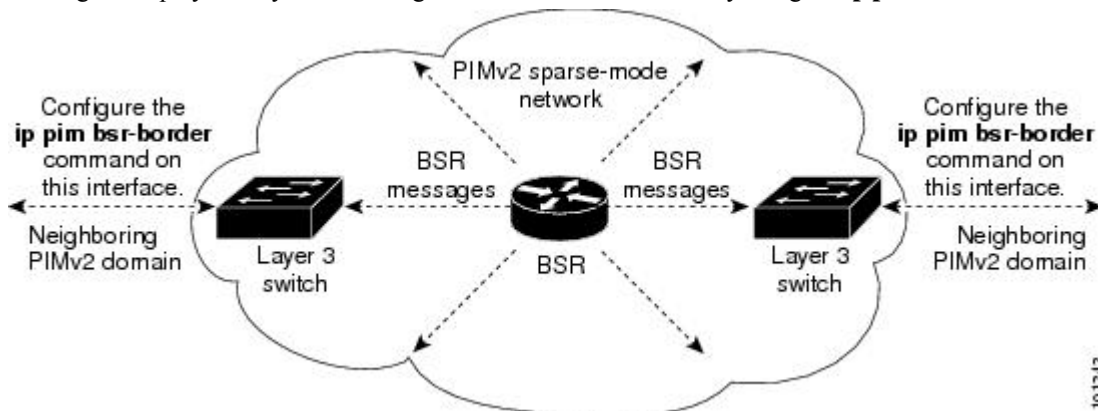
Benefits of Auto-RP in a PIM Network

- Auto-RP allows any change to the RP designation to be configured only on the devices that are RPs, not on the leaf routers.
- Auto-RP offers the ability to scope the RP address within a domain.

PIM Domain Border

As IP multicast becomes more widespread, the chance of one PIMv2 domain bordering another PIMv2 domain increases. Because two domains probably do not share the same set of RPs, BSR, candidate RPs, and candidate BSRs, you need to constrain PIMv2 BSR messages from flowing into or out of the domain. Allowing messages to leak across the domain borders could adversely affect the normal BSR election mechanism and elect a single BSR across all bordering domains and coningle candidate RP advertisements, resulting in the election of RPs in the wrong domain.

This figure displays how you can configure the PIM domain border by using the `ip pim bsr-border` command.



PIMv2 Bootstrap Router

PIMv2 Bootstrap Router (BSR) is another method to distribute group-to-RP mapping information to all PIM routers and multilayer devices in the network. It eliminates the need to manually configure RP information in every router and switch in the network. However, instead of using IP multicast to distribute group-to-RP mapping information, BSR uses hop-by-hop flooding of special BSR messages to distribute the mapping information.

The BSR is elected from a set of candidate routers and switches in the domain that have been configured to function as BSRs. The election mechanism is similar to the root-bridge election mechanism used in bridged LANs. The BSR election is based on the BSR priority of the device contained in the BSR messages that are sent hop-by-hop through the network. Each BSR device examines the message and forwards out all interfaces only the message that has either a higher BSR priority than its BSR priority or the same BSR priority, but with a higher BSR IP address. Using this method, the BSR is elected.

The elected BSR sends BSR messages with a TTL of 1. Neighboring PIMv2 routers or multilayer devices receive the BSR message and multicast it out all other interfaces (except the one on which it was received) with a TTL of 1. In this way, BSR messages travel hop-by-hop throughout the PIM domain. Because BSR messages contain the IP address of the current BSR, the flooding mechanism enables candidate RPs to automatically learn which device is the elected BSR.

Candidate RPs send candidate RP advertisements showing the group range for which they are responsible to the BSR, which stores this information in its local candidate-RP cache. The BSR periodically advertises the contents of this cache in BSR messages to all other PIM devices in the domain. These messages travel hop-by-hop through the network to all routers and switches, which store the RP information in the BSR message in their local RP cache. The routers and switches select the same RP for a given group because they all use a common RP hashing algorithm.

Multicast Forwarding

Forwarding of multicast traffic is accomplished by multicast-capable routers. These routers create distribution trees that control the path that IP multicast traffic takes through the network in order to deliver traffic to all receivers.

Multicast traffic flows from the source to the multicast group over a distribution tree that connects all of the sources to all of the receivers in the group. This tree may be shared by all sources (a shared tree) or a separate distribution tree can be built for each source (a source tree). The shared tree may be one-way or bidirectional.

Before describing the structure of source and shared trees, it is helpful to explain the notations that are used in multicast routing tables. These notations include the following:

- (S,G) = (unicast source for the multicast group G, multicast group G)
- (*,G) = (any source for the multicast group G, multicast group G)

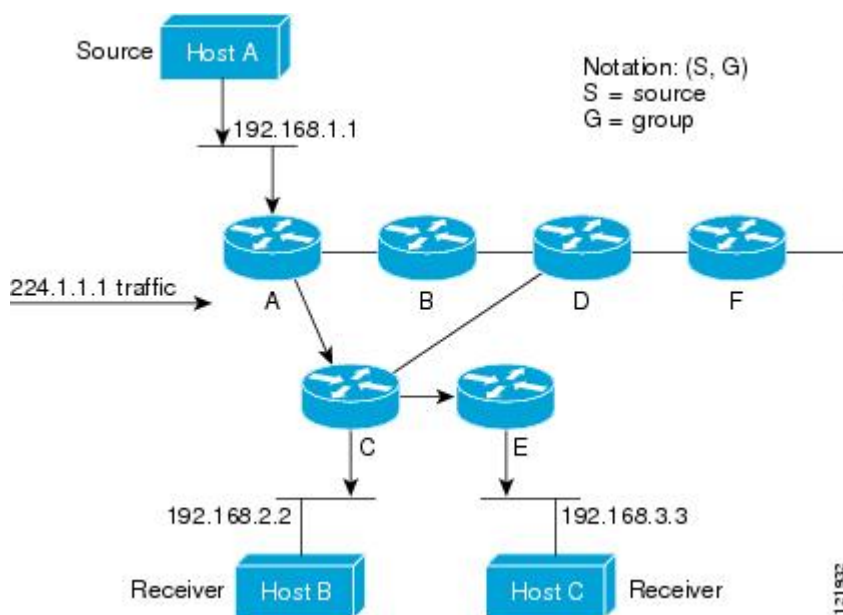
The notation of (S,G), pronounced “S comma G,” enumerates a shortest path tree where S is the IP address of the source and G is the multicast group address.

Shared trees are (*,G) and the source trees are (S,G) and always routed at the sources.

Multicast Distribution Source Tree

The simplest form of a multicast distribution tree is a source tree. A source tree has its root at the source host and has branches forming a spanning tree through the network to the receivers. Because this tree uses the shortest path through the network, it is also referred to as a shortest path tree (SPT).

The figure shows an example of an SPT for group 224.1.1.1 rooted at the source, Host A, and connecting two receivers, Hosts B and C.



Using standard notation, the SPT for the example shown in the figure would be (192.168.1.1, 224.1.1.1).

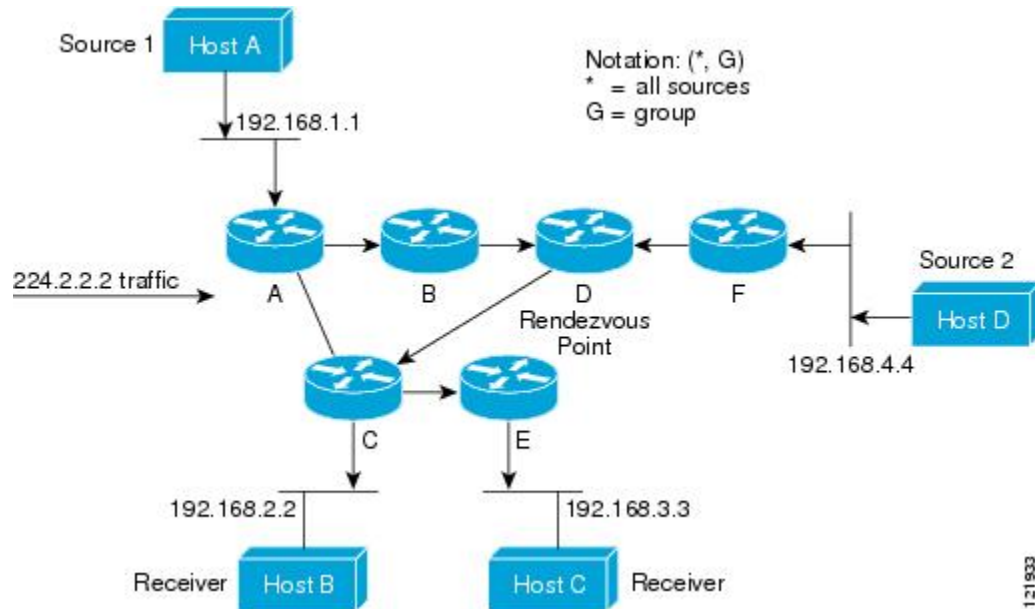
The (S,G) notation implies that a separate SPT exists for each individual source sending to each group--which is correct.

Multicast Distribution Shared Tree

Unlike source trees that have their root at the source, shared trees use a single common root placed at some chosen point in the network. This shared root is called a rendezvous point (RP).

The following figure shows a shared tree for the group 224.2.2.2 with the root located at Router D. This shared tree is unidirectional. Source traffic is sent towards the RP on a source tree. The traffic is then forwarded down the shared tree from the RP to reach all of the receivers (unless the receiver is located between the source and the RP, in which case it will be serviced directly).

Figure 12: Shared Tree



In this example, multicast traffic from the sources, Hosts A and D, travels to the root (Router D) and then down the shared tree to the two receivers, Hosts B and C. Because all sources in the multicast group use a common shared tree, a wildcard notation written as (*, G), pronounced "star comma G", represents the tree. In this case, * means all sources, and G represents the multicast group. Therefore, the shared tree shown in the figure would be written as (*, 224.2.2.2).

Both source trees and shared trees are loop-free. Messages are replicated only where the tree branches. Members of multicast groups can join or leave at any time; therefore the distribution trees must be dynamically updated. When all the active receivers on a particular branch stop requesting the traffic for a particular multicast group, the routers prune that branch from the distribution tree and stop forwarding traffic down that branch. If one receiver on that branch becomes active and requests the multicast traffic, the router will dynamically modify the distribution tree and start forwarding traffic again.

Source Tree Advantage

Source trees have the advantage of creating the optimal path between the source and the receivers. This advantage guarantees the minimum amount of network latency for forwarding multicast traffic. However, this optimization comes at a cost. The routers must maintain path information for each source. In a network that has thousands of sources and thousands of groups, this overhead can quickly become a resource issue on the routers. Memory consumption from the size of the multicast routing table is a factor that network designers must take into consideration.

Shared Tree Advantage

Shared trees have the advantage of requiring the minimum amount of state in each router. This advantage lowers the overall memory requirements for a network that only allows shared trees. The disadvantage of shared trees is that under certain circumstances the paths between the source and receivers might not be the optimal paths, which might introduce some latency in packet delivery. For example, in the figure above the shortest path between Host A (source 1) and Host B (a receiver) would be Router A and Router C. Because we are using Router D as the root for a shared tree, the traffic must traverse Routers A, B, D and then C.

Network designers must carefully consider the placement of the rendezvous point (RP) when implementing a shared tree-only environment.

In unicast routing, traffic is routed through the network along a single path from the source to the destination host. A unicast router does not consider the source address; it considers only the destination address and how to forward the traffic toward that destination. The router scans through its routing table for the destination address and then forwards a single copy of the unicast packet out the correct interface in the direction of the destination.

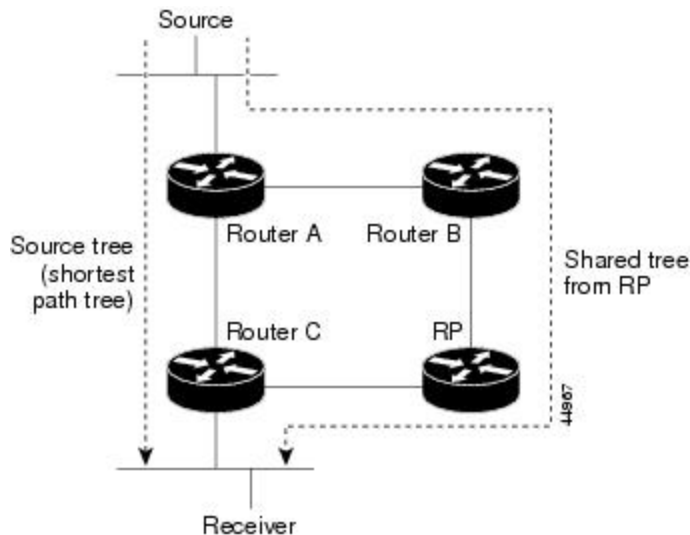
In multicast forwarding, the source is sending traffic to an arbitrary group of hosts that are represented by a multicast group address. The multicast router must determine which direction is the upstream direction (toward the source) and which one is the downstream direction (or directions) toward the receivers. If there are multiple downstream paths, the router replicates the packet and forwards it down the appropriate downstream paths (best unicast route metric)--which is not necessarily all paths. Forwarding multicast traffic away from the source, rather than to the receiver, is called Reverse Path Forwarding (RPF). RPF is described in the following section.

PIM Shared Tree and Source Tree

By default, members of a group receive data from senders to the group across a single data-distribution tree rooted at the RP.

Figure 13: Shared Tree and Source Tree (Shortest-Path Tree)

The following figure shows this type of shared-distribution tree. Data from senders is delivered to the RP for distribution to group members joined to the shared tree.



If the data rate warrants, leaf routers (routers without any downstream connections) on the shared tree can use the data distribution tree rooted at the source. This type of distribution tree is called a shortest-path tree or source tree. By default, the software device to a source tree upon receiving the first data packet from a source.

This process describes the move from a shared tree to a source tree:

1. A receiver joins a group; leaf Router C sends a join message toward the RP.
2. The RP puts a link to Router C in its outgoing interface list.
3. A source sends data; Router A encapsulates the data in a register message and sends it to the RP.

4. The RP forwards the data down the shared tree to Router C and sends a join message toward the source. At this point, data might arrive twice at Router C, once encapsulated and once natively.
5. When data arrives natively (unencapsulated) at the RP, it sends a register-stop message to Router A.
6. By default, reception of the first data packet prompts Router C to send a join message toward the source.
7. When Router C receives data on (S, G), it sends a prune message for the source up the shared tree.
8. The RP deletes the link to Router C from the outgoing interface of (S, G). The RP triggers a prune message toward the source.

Join and prune messages are sent for sources and RPs. They are sent hop-by-hop and are processed by each PIM device along the path to the source or RP. Register and register-stop messages are not sent hop-by-hop. They are sent by the designated router that is directly connected to a source and are received by the RP for the group.

Multiple sources sending to groups use the shared tree. You can configure the PIM device to stay on the shared tree.

The change from shared to source tree happens when the first data packet arrives at the last-hop router. This change depends upon the threshold that is configured by using the **ip pim spt-threshold** global configuration command.

The shortest-path tree requires more memory than the shared tree but reduces delay. You may want to postpone its use. Instead of allowing the leaf router to immediately move to the shortest-path tree, you can specify that the traffic must first reach a threshold.

You can configure when a PIM leaf router should join the shortest-path tree for a specified group. If a source sends at a rate greater than or equal to the specified kbps rate, the multilayer switch triggers a PIM join message toward the source to construct a source tree (shortest-path tree). If the traffic rate from the source drops below the threshold value, the leaf router switches back to the shared tree and sends a prune message toward the source.

You can specify to which groups the shortest-path tree threshold applies by using a group list (a standard access list). If a value of 0 is specified or if the group list is not used, the threshold applies to all groups.

Reverse Path Forwarding

In unicast routing, traffic is routed through the network along a single path from the source to the destination host. A unicast router does not consider the source address; it considers only the destination address and how to forward the traffic toward that destination. The router scans through its routing table for the destination network and then forwards a single copy of the unicast packet out the correct interface in the direction of the destination.

In multicast forwarding, the source is sending traffic to an arbitrary group of hosts that are represented by a multicast group address. The multicast router must determine which direction is the upstream direction (toward the source) and which one is the downstream direction (or directions) toward the receivers. If there are multiple downstream paths, the router replicates the packet and forwards it down the appropriate downstream paths (best unicast route metric)--which is not necessarily all paths. Forwarding multicast traffic away from the source, rather than to the receiver, is called Reverse Path Forwarding (RPF). RPF is an algorithm used for forwarding multicast datagrams.

Protocol Independent Multicast (PIM) uses the unicast routing information to create a distribution tree along the reverse path from the receivers towards the source. The multicast routers then forward packets along the distribution tree from the source to the receivers. RPF is a key concept in multicast forwarding. It enables

routers to correctly forward multicast traffic down the distribution tree. RPF makes use of the existing unicast routing table to determine the upstream and downstream neighbors. A router will forward a multicast packet only if it is received on the upstream interface. This RPF check helps to guarantee that the distribution tree will be loop-free.

RPF Check

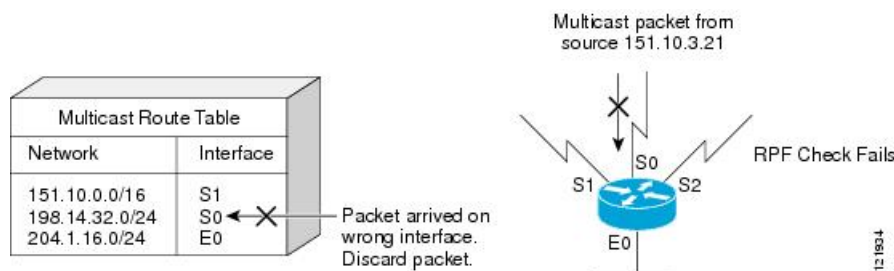
When a multicast packet arrives at a router, the router performs an RPF check on the packet. If the RPF check succeeds, the packet is forwarded. Otherwise, it is dropped.

For traffic flowing down a source tree, the RPF check procedure works as follows:

1. The router looks up the source address in the unicast routing table to determine if the packet has arrived on the interface that is on the reverse path back to the source.
2. If the packet has arrived on the interface leading back to the source, the RPF check succeeds and the packet is forwarded out the interfaces present in the outgoing interface list of a multicast routing table entry.
3. If the RPF check in Step 2 fails, the packet is dropped.

The figure shows an example of an unsuccessful RPF check.

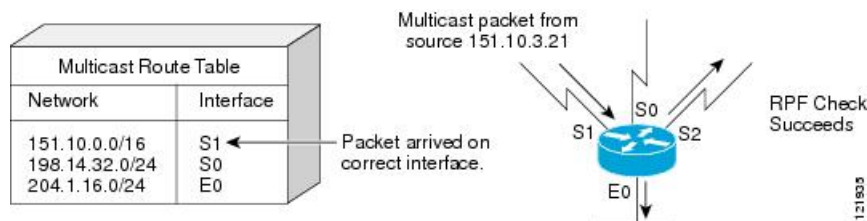
Figure 14: RPF Check Fails



As the figure illustrates, a multicast packet from source 151.10.3.21 is received on serial interface 0 (S0). A check of the unicast route table shows that S1 is the interface this router would use to forward unicast data to 151.10.3.21. Because the packet has arrived on interface S0, the packet is discarded.

The figure shows an example of a successful RPF check.

Figure 15: RPF Check Succeeds



In this example, the multicast packet has arrived on interface S1. The router refers to the unicast routing table and finds that S1 is the correct interface. The RPF check passes, and the packet is forwarded.

PIM uses both source trees and RP-rooted shared trees to forward datagrams. The RPF check is performed differently for each:

- If a PIM router or multilayer switch has a source-tree state (that is, an (S, G) entry is present in the multicast routing table), it performs the RPF check against the IP address of the source of the multicast packet.
- If a PIM router or multilayer switch has a shared-tree state (and no explicit source-tree state), it performs the RPF check on the RP address (which is known when members join the group).



Note DVMRP is not supported on the switch.

Sparse-mode PIM uses the RPF lookup function to decide where it needs to send joins and prunes:

- (S, G) joins (which are source-tree states) are sent toward the source.
- (*,G) joins (which are shared-tree states) are sent toward the RP.

Default PIM Routing Configuration

This table displays the default PIM routing configuration for the device.

Table 14: Default Multicast Routing Configuration

| Feature | Default Setting |
|-----------------------------------|-----------------------------|
| Multicast routing | Disabled on all interfaces. |
| PIM version | Version 2. |
| PIM mode | No mode is defined. |
| PIM stub routing | None configured. |
| PIM RP address | None configured. |
| PIM domain border | Disabled. |
| PIM multicast boundary | None. |
| Candidate BSRs | Disabled. |
| Candidate RPs | Disabled. |
| Shortest-path tree threshold rate | 0 kb/s. |
| PIM router query message interval | 30 seconds. |

How to Configure PIM

Enabling PIM Stub Routing

This procedure is optional.

Procedure

| | Command or Action | Purpose |
|---------------|--|--|
| Step 1 | enable Example: Device> enable | Enables privileged EXEC mode. Enter your password if prompted. |
| Step 2 | configure terminal Example: Device# configure terminal | Enters global configuration mode. |
| Step 3 | interface <i>interface-id</i> Example: Device (config)# interface gigabitethernet 1/0/1 | Specifies the interface on which you want to enable PIM stub routing, and enters interface configuration mode. <ul style="list-style-type: none"> • A routed port—A physical port that has been configured as a Layer 3 port by entering the no switchport interface configuration command. • An SVI—A VLAN interface created by using the interface vlan <i>vlan-id</i> global configuration command. |
| Step 4 | ip pim passive Example: Device (config-if)# ip pim passive | Configures the PIM stub feature on the interface. |
| Step 5 | end Example: Device (config)# end | Returns to privileged EXEC mode. |
| Step 6 | show ip pim interface Example: | (Optional) Displays the PIM stub that is enabled on each interface. |

| | Command or Action | Purpose |
|----------------|---|--|
| | Device# <code>show ip pim interface</code> | |
| Step 7 | show ip igmp groups detail Example: Device# <code>show ip igmp groups detail</code> | (Optional) Displays the interested clients that have joined the specific multicast source group. |
| Step 8 | show ip mroute Example: Device# <code>show ip mroute</code> | (Optional) Displays the IP multicast routing table. |
| Step 9 | show running-config Example: Device# <code>show running-config</code> | Verifies your entries. |
| Step 10 | copy running-config startup-config Example: Device# <code>copy running-config startup-config</code> | (Optional) Saves your entries in the configuration file. |

Configuring a Rendezvous Point

You must have a rendezvous point (RP), if the interface is in sparse-dense mode and if you want to handle the group as a sparse group. You can use these methods:

- By manually assigning an RP to multicast groups.
- As a standalone, Cisco-proprietary protocol separate from PIMv1, which includes:
- By using a standards track protocol in the Internet Engineering Task Force (IETF), which includes configuring PIMv2 BSR .



Note You can use Auto-RP, BSR, or a combination of both, depending on the PIM version that you are running and the types of routers in your network. For information about working with different PIM versions in your network, see [PIMv1 and PIMv2 Interoperability, on page 120](#).

Manually Assigning an RP to Multicast Groups

If the rendezvous point (RP) for a group is learned through a dynamic mechanism (such as Auto-RP or BSR), you need not perform this task for that RP.

Senders of multicast traffic announce their existence through register messages received from the source first-hop router (designated router) and forwarded to the RP. Receivers of multicast packets use RPs to join a multicast group by using explicit join messages.



Note RPs are not members of the multicast group; they serve as a *meeting place* for multicast sources and group members.

You can configure a single RP for multiple groups defined by an access list. If there is no RP configured for a group, the multilayer switch responds to the group as dense and uses the dense-mode PIM techniques.

This procedure is optional.

Procedure

| | Command or Action | Purpose |
|---------------|--|---|
| Step 1 | enable Example: Device> enable | Enables privileged EXEC mode. Enter your password if prompted. |
| Step 2 | configure terminal Example: Device# configure terminal | Enters global configuration mode. |
| Step 3 | ip pim rp-address <i>ip-address</i> <i>[access-list-number] [override]</i> Example: Device(config)# ip pim rp-address 10.1.1.1 20 override | Configures the address of a PIM RP. By default, no PIM RP address is configured. You must configure the IP address of RPs on all routers and multilayer switches (including the RP). Note If there is no RP configured for a group, the device treats the group as dense, using the dense-mode PIM techniques. A PIM device can be an RP for more than one group. Only one RP address can be used at a time within a PIM domain. The access list conditions specify for which groups the device is an RP. <ul style="list-style-type: none"> For <i>ip-address</i>, enter the unicast address of the RP in dotted-decimal notation. |

| | Command or Action | Purpose |
|---------------|---|--|
| | | <ul style="list-style-type: none"> • (Optional) For <i>access-list-number</i>, enter an IP standard access list number from 1 to 99. If no access list is configured, the RP is used for all groups. • (Optional) The override keyword indicates that if there is a conflict between the RP configured with this command and one learned by Auto-RP or BSR, the RP configured with this command prevails. |
| Step 4 | <p>access-list <i>access-list-number</i> {deny permit} <i>source</i> [<i>source-wildcard</i>]</p> <p>Example:</p> <pre>Device(config)# access-list 25 permit 10.5.0.1 255.224.0.0</pre> | <p>Creates a standard access list, repeating the command as many times as necessary.</p> <ul style="list-style-type: none"> • For <i>access-list-number</i>, enter the access list number specified in Step 2. • The deny keyword denies access if the conditions are matched. • The permit keyword permits access if the conditions are matched. • For <i>source</i>, enter the multicast group address for which the RP should be used. • (Optional) For <i>source-wildcard</i>, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. <p>The access list is always terminated by an implicit deny statement for everything.</p> |
| Step 5 | <p>end</p> <p>Example:</p> <pre>Device(config)# end</pre> | Returns to privileged EXEC mode. |
| Step 6 | <p>show running-config</p> <p>Example:</p> <pre>Device# show running-config</pre> | Verifies your entries. |
| Step 7 | <p>copy running-config startup-config</p> <p>Example:</p> <pre>Device# copy running-config startup-config</pre> | (Optional) Saves your entries in the configuration file. |

Setting Up Auto-RP in a New Internetwork



Note Omit Step 3 in the following procedure, if you want to configure a PIM router as the RP for the local group.

Procedure

| | Command or Action | Purpose |
|---------------|--|---|
| Step 1 | enable Example: Device> enable | Enables privileged EXEC mode. Enter your password if prompted. |
| Step 2 | show running-config Example: Device# show running-config | Verifies that a default RP is already configured on all PIM devices and the RP in the sparse-mode network. It was previously configured with the ip pim rp-address global configuration command. Note This step is not required for spare-dense-mode environments. The selected RP should have good connectivity and be available across the network. Use this RP for the global groups (for example, 224.x.x.x and other global groups). Do not reconfigure the group address range that this RP serves. RPs dynamically discovered through Auto-RP take precedence over statically configured RPs. Assume that it is desirable to use a second RP for the local groups. |
| Step 3 | configure terminal Example: Device# configure terminal | Enters global configuration mode. |
| Step 4 | ip pim send-rp-announce interface-id scope ttl group-list access-list-number interval seconds Example: Device (config)# ip pim send-rp-announce gigabitethernet 1/0/5 scope 20 group-list 10 interval 120 | Configures another PIM device to be the candidate RP for local groups. <ul style="list-style-type: none"> For <i>interface-id</i>, enter the interface type and number that identifies the RP address. Valid interfaces include physical ports, port channels, and VLANs. For scope ttl, specify the time-to-live value in hops. Enter a hop count that is |

| | Command or Action | Purpose |
|----------------------|--|--|
| | | <p>high enough so that the RP-announce messages reach all mapping agents in the network. There is no default setting. The range is 1 to 255.</p> <ul style="list-style-type: none"> For group-list <i>access-list-number</i>, enter an IP standard access list number from 1 to 99. If no access list is configured, the RP is used for all groups. For interval <i>seconds</i>, specify how often the announcement messages must be sent. The default is 60 seconds. The range is 1 to 16383. |
| <p>Step 5</p> | <p>access-list <i>access-list-number</i> {deny permit} <i>source</i> [<i>source-wildcard</i>]</p> <p>Example:</p> <pre>Device(config)# access-list 10 permit 10.10.0.0</pre> | <p>Creates a standard access list, repeating the command as many times as necessary.</p> <ul style="list-style-type: none"> For <i>access-list-number</i>, enter the access list number specified in Step 3. The deny keyword denies access if the conditions are matched. The permit keyword permits access if the conditions are matched. For <i>source</i>, enter the multicast group address range for which the RP should be used. (Optional) For <i>source-wildcard</i>, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. <p>Note Recall that the access list is always terminated by an implicit deny statement for everything.</p> |
| <p>Step 6</p> | <p>ip pim send-rp-discovery scope <i>ttl</i></p> <p>Example:</p> <pre>Device(config)# ip pim send-rp-discovery scope 50</pre> | <p>Finds a device whose connectivity is not likely to be interrupted, and assign it the role of RP-mapping agent.</p> <p>For scope <i>ttl</i>, specify the time-to-live value in hops to limit the RP discovery packets. All devices within the hop count from the source device receive the Auto-RP discovery messages. These messages tell other devices which group-to-RP mapping to use to avoid conflicts (such as overlapping group-to-RP</p> |

| | Command or Action | Purpose |
|----------------|---|--|
| | | ranges). There is no default setting. The range is 1 to 255. |
| Step 7 | end Example: Device (config) # end | Returns to privileged EXEC mode. |
| Step 8 | show running-config Example: Device# show running-config | Verifies your entries. |
| Step 9 | show ip pim rp mapping Example: Device# show ip pim rp mapping | Displays active RPs that are cached with associated multicast routing entries. |
| Step 10 | show ip pim rp Example: Device# show ip pim rp | Displays the information cached in the routing table. |
| Step 11 | copy running-config startup-config Example: Device# copy running-config startup-config | (Optional) Saves your entries in the configuration file. |

Adding Auto-RP to an Existing Sparse-Mode Cloud

This section contains suggestions for the initial deployment of Auto-RP into an existing sparse-mode cloud to minimize disruption of the existing multicast infrastructure.

This procedure is optional.

Procedure

| | Command or Action | Purpose |
|---------------|---|---|
| Step 1 | enable Example: Device> enable | Enables privileged EXEC mode. Enter your password if prompted. |

| | Command or Action | Purpose |
|---------------|---|--|
| Step 2 | <p>show running-config</p> <p>Example:</p> <pre>Device# show running-config</pre> | <p>Verifies that a default RP is already configured on all PIM devices and the RP in the sparse-mode network. It was previously configured with the ip pim rp-address global configuration command.</p> <p>Note This step is not required for sparse-dense-mode environments.</p> <p>The selected RP should have good connectivity and be available across the network. Use this RP for the global groups (for example, 224.x.x.x and other global groups). Do not reconfigure the group address range that this RP serves. RPs dynamically discovered through Auto-RP take precedence over statically configured RPs. Assume that it is desirable to use a second RP for the local groups.</p> |
| Step 3 | <p>configure terminal</p> <p>Example:</p> <pre>Device# configure terminal</pre> | <p>Enters global configuration mode.</p> |
| Step 4 | <p>ip pim send-rp-announce interface-id scope ttl group-list access-list-number interval seconds</p> <p>Example:</p> <pre>Device(config)# ip pim send-rp-announce gigabitethernet 1/0/5 scope 20 group-list 10 interval 120</pre> | <p>Configures another PIM device to be the candidate RP for local groups.</p> <ul style="list-style-type: none"> For <i>interface-id</i>, enter the interface type and number that identifies the RP address. Valid interfaces include physical ports, port channels, and VLANs. For scope <i>ttl</i>, specify the time-to-live value in hops. Enter a hop count that is high enough so that the RP-announce messages reach all mapping agents in the network. There is no default setting. The range is 1 to 255. For group-list <i>access-list-number</i>, enter an IP standard access list number from 1 to 99. If no access list is configured, the RP is used for all groups. For interval <i>seconds</i>, specify how often the announcement messages must be sent. The default is 60 seconds. The range is 1 to 16383. |

| | Command or Action | Purpose |
|---------------|--|--|
| Step 5 | <p>access-list <i>access-list-number</i> {deny permit} <i>source</i> [<i>source-wildcard</i>]</p> <p>Example:</p> <pre>Device (config)# access-list 10 permit 224.0.0.0 15.255.255.255</pre> | <p>Creates a standard access list, repeating the command as many times as necessary.</p> <ul style="list-style-type: none"> For <i>access-list-number</i>, enter the access list number specified in Step 3. The deny keyword denies access if the conditions are matched. The permit keyword permits access if the conditions are matched. For <i>source</i>, enter the multicast group address range for which the RP should be used. (Optional) For <i>source-wildcard</i>, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. <p>Recall that the access list is always terminated by an implicit deny statement for everything.</p> |
| Step 6 | <p>ip pim send-rp-discovery scope <i>tll</i></p> <p>Example:</p> <pre>Device (config)# ip pim send-rp-discovery scope 50</pre> | <p>Finds a device whose connectivity is not likely to be interrupted, and assigns it the role of RP-mapping agent.</p> <p>For scope <i>tll</i>, specify the time-to-live value in hops to limit the RP discovery packets. All devices within the hop count from the source device receive the Auto-RP discovery messages. These messages tell other devices which group-to-RP mapping to use to avoid conflicts (such as overlapping group-to-RP ranges). There is no default setting. The range is 1 to 255.</p> <p>Note To remove the device as the RP-mapping agent, use the no ip pim send-rp-discovery global configuration command.</p> |
| Step 7 | <p>end</p> <p>Example:</p> <pre>Device (config)# end</pre> | <p>Returns to privileged EXEC mode.</p> |
| Step 8 | <p>show running-config</p> <p>Example:</p> | <p>Verifies your entries.</p> |

| | Command or Action | Purpose |
|----------------|---|--|
| | Device# <code>show running-config</code> | |
| Step 9 | show ip pim rp mapping Example: Device# <code>show ip pim rp mapping</code> | Displays active RPs that are cached with associated multicast routing entries. |
| Step 10 | show ip pim rp Example: Device# <code>show ip pim rp</code> | Displays the information cached in the routing table. |
| Step 11 | copy running-config startup-config Example: Device# <code>copy running-config startup-config</code> | (Optional) Saves your entries in the configuration file. |

Preventing Join Messages to False RPs

Determine whether the `ip pim accept-rp` command was previously configured throughout the network by using the `show running-config` privileged EXEC command. If the `ip pim accept-rp` command is not configured on any device, this problem can be addressed later. In those routers or multilayer switches already configured with the `ip pim accept-rp` command, you must enter the command again to accept the newly advertised RP.

Filtering Incoming RP Announcement Messages

You can add configuration commands to the mapping agents to prevent a maliciously configured router from masquerading as a candidate RP and causing problems.

This procedure is optional.

Procedure

| | Command or Action | Purpose |
|---------------|---|---|
| Step 1 | enable Example: Device> <code>enable</code> | Enables privileged EXEC mode. Enter your password if prompted. |
| Step 2 | configure terminal Example: | Enters global configuration mode. |

| | Command or Action | Purpose |
|---------------|---|--|
| | Device# <code>configure terminal</code> | |
| Step 3 | <p>ip pim rp-announce-filter rp-list access-list-number group-list access-list-number</p> <p>Example:</p> <pre>Device(config)# ip pim rp-announce-filter rp-list 10 group-list 14</pre> | <p>Filters incoming RP announcement messages.</p> <p>Enter this command on each mapping agent in the network. Without this command, all incoming RP-announce messages are accepted by default.</p> <p>For rp-list access-list-number, configure an access list of candidate RP addresses that, if permitted, is accepted for the group ranges supplied in the group-list access-list-number variable. If this variable is omitted, the filter applies to all multicast groups.</p> <p>If more than one mapping agent is used, the filters must be consistent across all mapping agents to ensure that no conflicts occur in the group-to-RP mapping information.</p> |
| Step 4 | <p>access-list access-list-number {deny permit} source [source-wildcard]</p> <p>Example:</p> <pre>Device(config)# access-list 10 permit 10.8.1.0 255.255.224.0</pre> | <p>Creates a standard access list, repeating the command as many times as necessary.</p> <ul style="list-style-type: none"> • For <i>access-list-number</i>, enter the access list number specified in Step 2. • The deny keyword denies access if the conditions are matched. • The permit keyword permits access if the conditions are matched. • Create an access list that specifies from which routers and multilayer switches the mapping agent accepts candidate RP announcements (rp-list ACL). • Create an access list that specifies the range of multicast groups from which to accept or deny (group-list ACL). • For <i>source</i>, enter the multicast group address range for which the RP should be used. • (Optional) For <i>source-wildcard</i>, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. <p>The access list is always terminated by an implicit deny statement for everything.</p> |

| | Command or Action | Purpose |
|---------------|---|--|
| Step 5 | end Example: Device(config)# end | Returns to privileged EXEC mode. |
| Step 6 | show running-config Example: Device# show running-config | Verifies your entries. |
| Step 7 | copy running-config startup-config Example: Device# copy running-config startup-config | (Optional) Saves your entries in the configuration file. |

Configuring PIMv2 BSR

The process for configuring PIMv2 BSR may involve the following optional tasks:

- Defining the PIM domain border
- Defining the IP multicast boundary
- Configuring candidate BSRs
- Configuring candidate RPs

Defining the PIM Domain Border

Perform the following steps to configure the PIM domain border. This procedure is optional.

Procedure

| | Command or Action | Purpose |
|---------------|---|--|
| Step 1 | enable Example: Device> enable | Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted. |
| Step 2 | configure terminal Example: | Enters global configuration mode. |

| | Command or Action | Purpose |
|---------------|---|--|
| | Device# <code>configure terminal</code> | |
| Step 3 | interface <i>interface-id</i> Example: Device(config)# <code>interface gigabitethernet 1/0/1</code> | Specifies the interface to be configured, and enters interface configuration mode. |
| Step 4 | ip pim bsr-border Example: Device(config-if)# <code>ip pim bsr-border</code> | Defines a PIM bootstrap message boundary for the PIM domain. Enter this command on each interface that connects to other bordering PIM domains. This command instructs the device to neither send nor receive PIMv2 BSR messages on this interface. Note To remove the PIM border, use the no ip pim bsr-border interface configuration command. |
| Step 5 | end Example: Device(config)# <code>end</code> | Returns to privileged EXEC mode. |
| Step 6 | show running-config Example: Device# <code>show running-config</code> | Verifies your entries. |
| Step 7 | copy running-config startup-config Example: Device# <code>copy running-config startup-config</code> | (Optional) Saves your entries in the configuration file. |

Defining the IP Multicast Boundary

You define a multicast boundary to prevent Auto-RP messages from entering the PIM domain. You create an access list to deny packets destined for 224.0.1.39 and 224.0.1.40, which carry Auto-RP information.

This procedure is optional.

Procedure

| | Command or Action | Purpose |
|---------------|---|---|
| Step 1 | enable Example: <pre>Device> enable</pre> | Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted. |
| Step 2 | configure terminal Example: <pre># configure terminal</pre> | Enters global configuration mode. |
| Step 3 | access-list <i>access-list-number</i> deny source [source-wildcard] Example: <pre>Device(config)# access-list 12 deny 224.0.1.39 access-list 12 deny 224.0.1.40</pre> | Creates a standard access list, repeating the command as many times as necessary. <ul style="list-style-type: none"> • For <i>access-list-number</i>, the range is 1 to 99. • The deny keyword denies access if the conditions are matched. • For <i>source</i>, enter multicast addresses 224.0.1.39 and 224.0.1.40, which carry Auto-RP information. • (Optional) For <i>source-wildcard</i>, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. <p>The access list is always terminated by an implicit deny statement for everything.</p> |
| Step 4 | interface <i>interface-id</i> Example: <pre>Device(config)# interface gigabitethernet 1/0/1</pre> | Specifies the interface to be configured, and enters interface configuration mode. |
| Step 5 | ip multicast boundary <i>access-list-number</i> Example: <pre>Device(config-if)# ip multicast boundary 12</pre> | Configures the boundary, specifying the access list you created in Step 2. |
| Step 6 | end Example: | Returns to privileged EXEC mode. |

| | Command or Action | Purpose |
|---------------|---|--|
| | Device(config)# end | |
| Step 7 | show running-config Example: Device# show running-config | Verifies your entries. |
| Step 8 | copy running-config startup-config Example: Device# copy running-config startup-config | (Optional) Saves your entries in the configuration file. |

Configuring Candidate BSRs

You can configure one or more candidate BSRs. The devices serving as candidate BSRs should have good connectivity to other devices and be in the backbone portion of the network.

This procedure is optional.

Procedure

| | Command or Action | Purpose |
|---------------|---|---|
| Step 1 | enable Example: Device> enable | Enables privileged EXEC mode. Enter your password if prompted. |
| Step 2 | configure terminal Example: Device# configure terminal | Enters global configuration mode. |
| Step 3 | ip pim bsr-candidate interface-id hash-mask-length [priority] Example: Device(config)# ip pim bsr-candidate gigabitethernet 1/0/3 28 100 | Configures your device to be a candidate BSR. <ul style="list-style-type: none"> • For <i>interface-id</i>, enter the interface on this device from which the BSR address is derived to make it a candidate. This interface must be enabled with PIM. Valid interfaces include physical ports, port channels, and VLANs. • For <i>hash-mask-length</i>, specify the mask length (32 bits maximum) that is to be ANDed with the group address before the |

| | Command or Action | Purpose |
|---------------|---|--|
| | | <p>hash function is called. All groups with the same seed hash correspond to the same RP. For example, if this value is 24, only the first 24 bits of the group addresses matter.</p> <ul style="list-style-type: none"> • (Optional) For <i>priority</i>, enter a number from 0 to 255. The BSR with the larger priority is preferred. If the priority values are the same, the device with the highest IP address is selected as the BSR. The default is 0. |
| Step 4 | <p>end</p> <p>Example:</p> <pre>Device(config)# end</pre> | Returns to privileged EXEC mode. |
| Step 5 | <p>show running-config</p> <p>Example:</p> <pre>Device# show running-config</pre> | Verifies your entries. |
| Step 6 | <p>copy running-config startup-config</p> <p>Example:</p> <pre>Device# copy running-config startup-config</pre> | (Optional) Saves your entries in the configuration file. |

Configuring the Candidate RPs

You can configure one or more candidate RPs. Similar to BSRs, the RPs should also have good connectivity to other devices and be in the backbone portion of the network. An RP can serve the entire IP multicast address space or a portion of it. Candidate RPs send candidate RP advertisements to the BSR.

This procedure is optional.

Before you begin

When deciding which devices should be RPs, consider these options:

- In a network of Cisco routers and multilayer switches where only Auto-RP is used, any device can be configured as an RP.
- In a network that includes only Cisco PIMv2 routers and multilayer switches and with routers from other vendors, any device can be used as an RP.

- In a network of Cisco PIMv1 routers, Cisco PIMv2 routers, and routers from other vendors, configure only Cisco PIMv2 routers and multilayer switches as RPs.

Procedure

| | Command or Action | Purpose |
|---------------|--|--|
| Step 1 | enable Example: Device> enable | Enables privileged EXEC mode. Enter your password if prompted. |
| Step 2 | configure terminal Example: Device# configure terminal | Enters global configuration mode. |
| Step 3 | ip pim rp-candidate <i>interface-id</i> [group-list <i>access-list-number</i>] Example: Device(config)# ip pim rp-candidate gigabitethernet 1/0/5 group-list 10 | Configures your device to be a candidate RP. <ul style="list-style-type: none"> • For <i>interface-id</i>, specify the interface whose associated IP address is advertised as a candidate RP address. Valid interfaces include physical ports, port channels, and VLANs. • (Optional) For group-list <i>access-list-number</i>, enter an IP standard access list number from 1 to 99. If no group-list is specified, the device is a candidate RP for all groups. |
| Step 4 | access-list <i>access-list-number</i> { deny permit } <i>source</i> [<i>source-wildcard</i>] Example: Device(config)# access-list 10 permit 239.0.0.0 0.255.255.255 | Creates a standard access list, repeating the command as many times as necessary. <ul style="list-style-type: none"> • For <i>access-list-number</i>, enter the access list number specified in Step 2. • The deny keyword denies access if the conditions are matched. The permit keyword permits access if the conditions are matched. • For <i>source</i>, enter the number of the network or host from which the packet is being sent. • (Optional) For <i>source-wildcard</i>, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. |

| | Command or Action | Purpose |
|---------------|---|--|
| | | The access list is always terminated by an implicit deny statement for everything. |
| Step 5 | end Example: Device(config)# end | Returns to privileged EXEC mode. |
| Step 6 | show running-config Example: Device# show running-config | Verifies your entries. |
| Step 7 | copy running-config startup-config Example: Device# copy running-config startup-config | (Optional) Saves your entries in the configuration file. |

Configuring Sparse Mode with Auto-RP

Before you begin

- All access lists that are needed when Auto-RP is configured should be configured prior to beginning the configuration task.



Note

- If a group has no known RP and the interface is configured to be sparse-dense mode, the interface is treated as if it were in dense mode, and data is flooded over the interface. To avoid this data flooding, configure the Auto-RP listener and then configure the interface as sparse mode.
- When configuring Auto-RP, you must either configure the Auto-RP listener feature (Step 5) and specify sparse mode (Step 7).
- When you configure sparse-dense mode, dense mode failover may result in a network dense-mode flood. To avoid this condition, use PIM sparse mode with the Auto-RP listener feature.

Follow this procedure to configure auto-rendezvous point (Auto-RP). Auto-RP can also be optionally used with anycast RP.

Procedure

| | Command or Action | Purpose |
|---------------|-------------------|-------------------------------|
| Step 1 | enable | Enables privileged EXEC mode. |

| | Command or Action | Purpose |
|---------------|--|--|
| | Example: Device> enable | <ul style="list-style-type: none"> • Enter your password if prompted. |
| Step 2 | configure terminal Example: Device# configure terminal | Enters global configuration mode. |
| Step 3 | ip multicast-routing Example: Device(config)# ip multicast-routing | Enables IP multicast routing. |
| Step 4 | Either perform Steps 5 through 7 or perform Steps 6 and 8. | -- |
| Step 5 | interface type number Example: Device(config)# interface GigabitEthernet 1/0/0 | Selects an interface that is connected to hosts on which PIM can be enabled. |
| Step 6 | ip pim sparse-mode Example: Device(config-if)# ip pim sparse-mode | Enables PIM sparse mode on an interface. When configuring Auto-RP in sparse mode, you must also configure the Auto-RP listener in the next step. <ul style="list-style-type: none"> • Skip this step if you are configuring sparse-dense mode in Step 8. |
| Step 7 | exit Example: Device(config-if)# exit | Exits interface configuration mode and returns to global configuration mode. |
| Step 8 | Repeat Steps 1 through 9 on all PIM interfaces. | -- |
| Step 9 | ip pim send-rp-announce { <i>interface-type interface-number</i> <i>ip-address</i> } scope <i>ttl-value</i> [group-list <i>access-list</i>] [interval <i>seconds</i>] [bidir] Example: Device(config)# ip pim send-rp-announce loopback0 scope 31 group-list 5 | Sends RP announcements out all PIM-enabled interfaces. <ul style="list-style-type: none"> • Perform this step on the RP device only. • Use the <i>interface-type</i> and <i>interface-number</i> arguments to define which IP address is to be used as the RP address. • Use the <i>ip-address</i> argument to specify a directly connected IP address as the RP address. |

| | Command or Action | Purpose |
|-----------------------|---|---|
| | | <p>Note If the <i>ip-address</i> argument is configured for this command, the RP-announce message will be sourced by the interface to which this IP address is connected (that is, the source address in the IP header of the RP-announce message is the IP address of that interface).</p> <ul style="list-style-type: none"> • This example shows that the interface is enabled with a maximum of 31 hops. The IP address by which the device wants to be identified as RP is the IP address associated with loopback interface 0. Access list 5 describes the groups for which this device serves as RP. |
| <p>Step 10</p> | <p>ip pim send-rp-discovery [<i>interface-type interface-number</i>] scope <i>tvl-value</i> [interval <i>seconds</i>]</p> <p>Example:</p> <pre>Device(config)# ip pim send-rp-discovery loopback 1 scope 31</pre> | <p>Configures the device to be an RP mapping agent.</p> <ul style="list-style-type: none"> • Perform this step on RP mapping agent devices or on combined RP/RP mapping agent devices. <p>Note Auto-RP allows the RP function to run separately on one device and the RP mapping agent to run on one or multiple devices. It is possible to deploy the RP and the RP mapping agent on a combined RP/RP mapping agent device.</p> <ul style="list-style-type: none"> • Use the optional <i>interface-type</i> and <i>interface-number</i> arguments to define which IP address is to be used as the source address of the RP mapping agent. • Use the scope keyword and <i>tvl-value</i> argument to specify the Time-to-Live (TTL) value in the IP header of Auto-RP discovery messages. • Use the optional interval keyword and <i>seconds</i> argument to specify the interval at which Auto-RP discovery messages are sent. |

| | Command or Action | Purpose |
|----------------|--|---|
| | | <p>Note Lowering the interval at which Auto-RP discovery messages are sent from the default value of 60 seconds results in more frequent floodings of the group-to-RP mappings. In some network environments, the disadvantages of lowering the interval (more control packet overhead) may outweigh the advantages (more frequent group-to-RP mapping updates).</p> <ul style="list-style-type: none"> • The example shows limiting the Auto-RP discovery messages to 31 hops on loopback interface 1. |
| Step 11 | <p>ip pim rp-announce-filter rp-list <i>access-list</i> group-list <i>access-list</i></p> <p>Example:</p> <pre>Device(config)# ip pim rp-announce-filter rp-list 1 group-list 2</pre> | <p>Filters incoming RP announcement messages sent from candidate RPs (C-RPs) to the RP mapping agent.</p> <ul style="list-style-type: none"> • Perform this step on the RP mapping agent only. |
| Step 12 | <p>interface <i>type number</i></p> <p>Example:</p> <pre>Device(config)# interface gigabitethernet 1/0/0</pre> | <p>Selects an interface that is connected to hosts on which PIM can be enabled.</p> |
| Step 13 | <p>ip multicast boundary <i>access-list</i> [filter-autorp]</p> <p>Example:</p> <pre>Device(config-if)# ip multicast boundary 10 filter-autorp</pre> | <p>Configures an administratively scoped boundary.</p> <ul style="list-style-type: none"> • Perform this step on the interfaces that are boundaries to other devices. • The access list is not shown in this task. • An access list entry that uses the deny keyword creates a multicast boundary for packets that match that entry. |
| Step 14 | <p>end</p> <p>Example:</p> <pre>Device(config-if)# end</pre> | <p>Returns to global configuration mode.</p> |
| Step 15 | <p>show ip pim autorp</p> <p>Example:</p> | <p>(Optional) Displays the Auto-RP information.</p> |

| | Command or Action | Purpose |
|----------------|--|---|
| | Device# show ip pim autorp | |
| Step 16 | show ip pim rp [mapping] [rp-address] Example: Device# show ip pim rp mapping | (Optional) Displays RPs known in the network and shows how the device learned about each RP. |
| Step 17 | show ip igmp groups [group-name group-address interface-type interface-number] [detail] Example: Device# show ip igmp groups | (Optional) Displays the multicast groups having receivers that are directly connected to the device and that were learned through Internet Group Management Protocol (IGMP). <ul style="list-style-type: none"> • A receiver must be active on the network at the time that this command is issued in order for receiver information to be present on the resulting display. |
| Step 18 | show ip mroute [group-address group-name] [source-address source-name] [interface-type interface-number] [summary] [count] [active kbps] Example: Device# show ip mroute cbone-audio | (Optional) Displays the contents of the IP multicast routing (mroute) table. |

Delaying the Use of PIM Shortest-Path Tree

Perform these steps to configure a traffic rate threshold that must be reached before multicast routing is switched from the source tree to the shortest-path tree.

This procedure is optional.

Procedure

| | Command or Action | Purpose |
|---------------|---|--|
| Step 1 | enable Example: Device> enable | Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted. |
| Step 2 | configure terminal Example: Device# configure terminal | Enters global configuration mode. |

| | Command or Action | Purpose |
|---------------|--|---|
| Step 3 | <p>access-list <i>access-list-number</i> {deny permit} <i>source</i> [<i>source-wildcard</i>]</p> <p>Example:</p> <pre>Device(config)# access-list 16 permit 225.0.0.0 0.255.255.255</pre> | <p>Creates a standard access list.</p> <ul style="list-style-type: none"> For <i>access-list-number</i>, the range is 1 to 99. The deny keyword denies access if the conditions are matched. The permit keyword permits access if the conditions are matched. For <i>source</i>, specify the multicast group to which the threshold will apply. (Optional) For <i>source-wildcard</i>, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. <p>The access list is always terminated by an implicit deny statement for everything.</p> |
| Step 4 | <p>ip pim spt-threshold {<i>kbps</i> infinity} [group-list <i>access-list-number</i>]</p> <p>Example:</p> <pre>Device(config)# ip pim spt-threshold infinity group-list 16</pre> | <p>Specifies the threshold that must be reached before moving to shortest-path tree (spt).</p> <ul style="list-style-type: none"> For <i>kbps</i>, specify the traffic rate in kilobits per second. The default is 0 kbps. <p>Note Because of device hardware limitations, 0 kbps is the only valid entry even though the range is 0 to 4294967.</p> <ul style="list-style-type: none"> Specify infinity if you want all sources for the specified group to use the shared tree, never switching to the source tree. (Optional) For group-list <i>access-list-number</i>, specify the access list created in Step 2. If the value is 0 or if the group list is not used, the threshold applies to all groups. |
| Step 5 | <p>end</p> <p>Example:</p> <pre>Device(config)# end</pre> | Returns to privileged EXEC mode. |
| Step 6 | <p>show running-config</p> <p>Example:</p> | Verifies your entries. |

| | Command or Action | Purpose |
|---------------|---|--|
| | Device# <code>show running-config</code> | |
| Step 7 | copy running-config startup-config Example: Device# <code>copy running-config startup-config</code> | (Optional) Saves your entries in the configuration file. |

Modifying the PIM Router-Query Message Interval

PIM routers and multilayer switches send PIM router-query messages to find which device will be the designated router (DR) for each LAN segment (subnet). The DR is responsible for sending IGMP host-query messages to all hosts on the directly connected LAN.

With PIM DM operation, the DR has meaning only if IGMPv1 is in use. IGMPv1 does not have an IGMP querier election process, so the elected DR functions as the IGMP querier. With PIM-SM operation, the DR is the device that is directly connected to the multicast source. It sends PIM register messages to notify the RP that multicast traffic from a source needs to be forwarded down the shared tree. In this case, the DR is the device with the highest IP address.

This procedure is optional.

Procedure

| | Command or Action | Purpose |
|---------------|---|--|
| Step 1 | enable Example: Device> <code>enable</code> | Enables privileged EXEC mode. Enter your password if prompted. |
| Step 2 | configure terminal Example: Device# <code>configure terminal</code> | Enters global configuration mode. |
| Step 3 | interface <i>interface-id</i> Example: Device(config)# <code>interface gigabitethernet 1/0/1</code> | Specifies the interface to be configured, and enters interface configuration mode. |
| Step 4 | ip pim query-interval <i>seconds</i> Example: | Configures the frequency at which the device sends PIM router-query messages. |

| | Command or Action | Purpose |
|---------------|---|--|
| | Device(config-if)# <code>ip pim query-interval 45</code> | The default is 30 seconds. The range is 1 to 65535. |
| Step 5 | end Example: Device(config)# <code>end</code> | Returns to privileged EXEC mode. |
| Step 6 | show ip igmp interface [<i>interface-id</i>] Example: Device# <code>show ip igmp interface</code> | Verifies your entries. |
| Step 7 | copy running-config startup-config Example: Device# <code>copy running-config startup-config</code> | (Optional) Saves your entries in the configuration file. |

Verifying PIM Operations

Verifying IP Multicast Operation in a PIM-SM or a PIM-SSM Network

When you verify the operation of IP multicast in a PIM-SM network environment or in a PIM-SSM network environment, a useful approach is to begin the verification process on the last hop router, and then continue the verification process on the routers along the SPT until the first hop router has been reached. The goal of the verification is to ensure that IP multicast traffic is being routed properly through an IP multicast network.

Perform the following optional tasks to verify IP multicast operation in a PIM-SM or a PIM-SSM network. The steps in these tasks help to locate a faulty hop when sources and receivers are not operating as expected.



Note If packets are not reaching their expected destinations, you might want consider disabling IP multicast fast switching, which would place the router in process switching mode. If packets begin reaching their proper destinations after IP multicast fast switching has been disabled, then the issue most likely was related to IP multicast fast switching.

Verifying IP Multicast on the First Hop Router

Enter these commands on the first hop router to verify IP multicast operations on the first hop router:

Procedure

| | Command or Action | Purpose |
|---------------|--|---|
| Step 1 | <p>enable</p> <p>Example:</p> <pre>Device> enable</pre> | <p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> • Enter your password if prompted. |
| Step 2 | <p>show ip mroute [group-address]</p> <p>Example:</p> <pre>Device# show ip mroute 239.1.2.3 (*, 239.1.2.3), 00:18:10/stopped, RP 172.16.0.1, flags: SPF Incoming interface: Serial1/0, RPF nbr 172.31.200.2 Outgoing interface list: Null (10.0.0.1, 239.1.2.3), 00:18:10/00:03:22, flags: FT Incoming interface: GigabitEthernet0/0/0, RPF nbr 0.0.0.0 Outgoing interface list: Serial1/0, Forward/Sparse, 00:18:10/00:03:19</pre> | <p>Confirms that the F flag has been set for mroutes on the first hop router.</p> |
| Step 3 | <p>show ip mroute active [kb/s]</p> <p>Example:</p> <pre>Device# show ip mroute active Active IP Multicast Sources - sending >= 4 kbps Group: 239.1.2.3, (?) Source: 10.0.0.1 (?) Rate: 20 pps/4 kbps(1sec), 4 kbps(last 30 secs), 4 kbps(life avg)</pre> | <p>Displays information about active multicast sources sending to groups. The output of this command provides information about the multicast packet rate for active sources.</p> <p>Note By default, the output of the show ip mroute command with the active keyword displays information about active sources sending traffic to groups at a rate greater than or equal to 4 kb/s. To display information about active sources sending low-rate traffic to groups (that is, traffic less than 4 kb/s), specify a value of 1 for the <i>kb/s</i> argument. Specifying a value of 1 for this argument displays information about active sources sending traffic to groups at a rate equal to or greater than 1 kb/s, which effectively displays information about all possible active source traffic.</p> |

Verifying IP Multicast on Routers Along the SPT

Enter these commands on routers along the SPT to verify IP multicast operations on routers along the SPT in a PIM-SM or PIM-SSM network:

Procedure

| | Command or Action | Purpose |
|---------------|--|--|
| Step 1 | enable Example: Device> enable | Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted. |
| Step 2 | show ip mroute [<i>group-address</i>] Example: Device# show ip mroute 239.1.2.3 (*, 239.1.2.3), 00:17:56/00:03:02, RP 172.16.0.1, flags: S Incoming interface: Null, RPF nbr 0.0.0.0 Outgoing interface list: GigabitEthernet0/0/0, Forward/Sparse, 00:17:56/00:03:02 (10.0.0.1, 239.1.2.3), 00:15:34/00:03:28, flags: T Incoming interface: Serial1/0, RPF nbr 172.31.200.1 Outgoing interface list: GigabitEthernet0/0/0, Forward/Sparse, 00:15:34/00:03:02 | Confirms the RPF neighbor towards the source for a particular group or groups. |
| Step 3 | show ip mroute active Example: Device# show ip mroute active Active IP Multicast Sources - sending >= 4 kbps | Displays information about active multicast sources sending to groups. The output of this command provides information about the multicast packet rate for active sources. |

| | Command or Action | Purpose |
|--|---|---|
| | <pre>Group: 239.1.2.3, (?) Source: 10.0.0.1 (?) Rate: 20 pps/4 kbps(1sec), 4 kbps(last 30 secs), 4 kbps(life avg)</pre> | <p>Note By default, the output of the show ip mroute command with the active keyword displays information about active sources sending traffic to groups at a rate greater than or equal to 4 kb/s. To display information about active sources sending low-rate traffic to groups (that is, traffic less than 4 kb/s), specify a value of 1 for the <i>kb/s</i> argument. Specifying a value of 1 for this argument displays information about active sources sending traffic to groups at a rate equal to or greater than 1 kb/s, which effectively displays information about all possible active source traffic.</p> |

Verifying IP Multicast Operation on the Last Hop Router

Enter these commands on the last hop router to verify IP multicast operations on the last hop router:

Procedure

| | Command or Action | Purpose |
|---------------|---|--|
| Step 1 | <p>enable</p> <p>Example:</p> <pre>Device> enable</pre> | <p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> • Enter your password if prompted. |
| Step 2 | <p>show ip igmp groups</p> <p>Example:</p> <pre>Device# show ip igmp groups IGMP Connected Group Membership Group Address Interface Uptime Expires Last Reporter 239.1.2.3 GigabitEthernet1/0/0 00:05:14 00:02:14 10.1.0.6 224.0.1.39 GigabitEthernet0/0/0 00:09:11 00:02:08 172.31.100.1</pre> | <p>Verifies IGMP memberships on the last hop router. This information will confirm the multicast groups with receivers that are directly connected to the last hop router and that are learned through IGMP.</p> |
| Step 3 | <p>show ip pim rp mapping</p> <p>Example:</p> | <p>Confirms that the group-to-RP mappings are being populated correctly on the last hop router.</p> |

| | Command or Action | Purpose |
|---------------|---|---|
| | <pre>Device# show ip pim rp mapping PIM Group-to-RP Mappings Group(s) 224.0.0.0/4 RP 172.16.0.1 (?), v2v1 Info source: 172.16.0.1 (?), elected via Auto-RP Uptime: 00:09:11, expires: 00:02:47</pre> | <p>Note Ignore this step if you are verifying a last hop router in a PIM-SSM network. The show ip pim rp mapping command does not work with routers in a PIM-SSM network because PIM-SSM does not use RPs. In addition, if configured correctly, PIM-SSM groups do not appear in the output of the show ip pim rp mapping command.</p> |
| Step 4 | <p>show ip mroute</p> <p>Example:</p> <pre>Device# show ip mroute (*, 239.1.2.3), 00:05:14/00:03:04, RP 172.16.0.1, flags: SJC Incoming interface: GigabitEthernet0/0/0, RPF nbr 172.31.100.1 Outgoing interface list: GigabitEthernet1/0, Forward/Sparse, 00:05:10/00:03:04 (10.0.0.1, 239.1.2.3), 00:02:49/00:03:29, flags: T Incoming interface: GigabitEthernet0/0/0, RPF nbr 172.31.100.1 Outgoing interface list: GigabitEthernet1/0, Forward/Sparse, 00:02:49/00:03:04 (*, 224.0.1.39), 00:10:05/stopped, RP 0.0.0.0, flags: DC Incoming interface: Null, RPF nbr 0.0.0.0 Outgoing interface list: GigabitEthernet1/0, Forward/Sparse, 00:05:15/00:00:00 GigabitEthernet0/0, Forward/Sparse, 00:10:05/00:00:00 (172.16.0.1, 224.0.1.39), 00:02:00/00:01:33, flags: PTX Incoming interface: GigabitEthernet0/0/0, RPF nbr 172.31.100.1</pre> | <p>Verifies that the mroute table is being populated properly on the last hop router.</p> |
| Step 5 | <p>show ip interface [type number]</p> <p>Example:</p> <pre>Device# show ip interface GigabitEthernet 0/0/0</pre> | <p>Verifies that multicast fast switching is enabled for optimal performance on the outgoing interface on the last hop router.</p> |

| | Command or Action | Purpose |
|---------------|---|---|
| | <pre>GigabitEthernet0/0/0 is up, line protocol is up Internet address is 172.31.100.2/24 Broadcast address is 255.255.255.255 Address determined by setup command MTU is 1500 bytes Helper address is not set Directed broadcast forwarding is disabled Multicast reserved groups joined: 224.0.0.1 224.0.0.22 224.0.0.13 224.0.0.5 224.0.0.6 Outgoing access list is not set Inbound access list is not set Proxy ARP is enabled Local Proxy ARP is disabled Security level is default Split horizon is enabled ICMP redirects are always sent ICMP unreachable are always sent ICMP mask replies are never sent IP fast switching is enabled IP fast switching on the same interface is disabled IP Flow switching is disabled IP CEF switching is disabled IP Fast switching turbo vector IP multicast fast switching is enabled IP route-cache flags are Fast Router Discovery is disabled IP output packet accounting is disabled IP access violation accounting is disabled TCP/IP header compression is disabled RTP/IP header compression is disabled Policy routing is disabled Network address translation is disabled WCCP Redirect outbound is disabled WCCP Redirect inbound is disabled WCCP Redirect exclude is disabled BGP Policy Mapping is disabled</pre> | <p>Note Using the no ip mroute-cache interface command disables IP multicast fast-switching. When IP multicast fast switching is disabled, packets are forwarded through the process-switched path.</p> |
| Step 6 | <p>show ip mfib</p> <p>Example:</p> <pre>Device# show ip mfib</pre> | <p>Displays the forwarding entries and interfaces in the IP Multicast Forwarding Information Base (MFIB).</p> |
| Step 7 | <p>show ip pim interface count</p> <p>Example:</p> <pre>Device# show ip pim interface count State: * - Fast Switched, H - Hardware Switching Enabled Address Interface FS Mpackets In/Out 172.31.100.2 GigabitEthernet0/0/0 * 4122/0</pre> | <p>Confirms that multicast traffic is being forwarded on the last hop router.</p> |

| | Command or Action | Purpose |
|---------------|---|---|
| | <pre>10.1.0.1 GigabitEthernet1/0/0 * 0/3193</pre> | |
| Step 8 | <p>show ip mroute count</p> <p>Example:</p> <pre>Device# show ip mroute count IP Multicast Statistics 6 routes using 4008 bytes of memory 3 groups, 1.00 average sources per group Forwarding Counts: Pkt Count/Pkts per second/Avg Pkt Size/Kilobits per second Other counts: Total/RPF failed/Other drops(OIF-null, rate-limit etc) Group: 239.1.2.3, Source count: 1, Packets forwarded: 3165, Packets received: 3165 RP-tree: Forwarding: 0/0/0/0, Other: 0/0/0 Source: 10.0.0.1/32, Forwarding: 3165/20/28/4, Other: 0/0/0 Group: 224.0.1.39, Source count: 1, Packets forwarded: 21, Packets received: 120 Source: 172.16.0.1/32, Forwarding: 21/1/48/0, Other: 120/0/99 Group: 224.0.1.40, Source count: 1, Packets forwarded: 10, Packets received: 10 Source: 172.16.0.1/32, Forwarding: 10/1/48/0, Other: 10/0/0</pre> | Confirms that multicast traffic is being forwarded on the last hop router. |
| Step 9 | <p>show ip mroute active [kb/s]</p> <p>Example:</p> <pre>Device# show ip mroute active Active IP Multicast Sources - sending >= 4 kbps</pre> | Displays information about active multicast sources sending traffic to groups on the last hop router. The output of this command provides information about the multicast packet rate for active sources. |

| | Command or Action | Purpose |
|--|---|---|
| | <pre>Group: 239.1.2.3, (?) Source: 10.0.0.1 (?) Rate: 20 pps/4 kbps(1sec), 4 kbps(last 50 secs), 4 kbps(life avg)</pre> | <p>Note By default, the output of the show ip mroute command with the active keyword displays information about active sources sending traffic to groups at a rate greater than or equal to 4 kb/s. To display information about active sources sending low-rate traffic to groups (that is, traffic less than 4 kb/s), specify a value of 1 for the <i>kb/s</i> argument. Specifying a value of 1 for this argument displays information about active sources sending traffic to groups at a rate equal to or greater than 1 kb/s, which effectively displays information about all possible active source traffic.</p> |

Using PIM-Enabled Routers to Test IP Multicast Reachability

If all the PIM-enabled routers and access servers that you administer are members of a multicast group, pinging that group causes all routers to respond, which can be a useful administrative and debugging tool.

To use PIM-enabled routers to test IP multicast reachability, perform the following tasks:

Configuring Routers to Respond to Multicast Pings

Follow these steps to configure a router to respond to multicast pings. Perform the task on all the interfaces of a router and on all the routers participating in the multicast network:

Procedure

| | Command or Action | Purpose |
|---------------|--|--|
| Step 1 | <pre>enable</pre> <p>Example:</p> <pre>Device> enable</pre> | Enables privileged EXEC mode. Enter your password if prompted. |
| Step 2 | <pre>configure terminal</pre> <p>Example:</p> <pre>Device# configure terminal</pre> | Enters global configuration mode. |
| Step 3 | <pre>interface type number</pre> <p>Example:</p> | Enters interface configuration mode. |

| | Command or Action | Purpose |
|---------------|--|--|
| | Device(config)# interface gigabitethernet 1/0/0 | For the <i>type</i> and <i>number</i> arguments, specify an interface that is directly connected to hosts or is facing hosts. |
| Step 4 | ip igmp join-group <i>group-address</i> Example: Device(config-if)# ip igmp join-group 225.2.2.2 | (Optional) Configures an interface on the router to join the specified group. For the purpose of this task, configure the same group address for the <i>group-address</i> argument on all interfaces on the router participating in the multicast network. Note With this method, the router accepts the multicast packets in addition to forwarding them. Accepting the multicast packets prevents the router from fast switching. |
| Step 5 | Repeat Step 3 and Step 4 for each interface on the router participating in the multicast network. | -- |
| Step 6 | end Example: Device(config-if)# end | Ends the current configuration session and returns to privileged EXEC mode. |

Pinging Routers Configured to Respond to Multicast Pings

Follow these steps on a router to initiate a ping test to the routers configured to respond to multicast pings. This task is used to test IP multicast reachability in a network.

Procedure

| | Command or Action | Purpose |
|---------------|--|--|
| Step 1 | enable Example: Device> enable | Enables privileged EXEC mode. Enter your password if prompted. |
| Step 2 | ping <i>group-address</i> Example: Device# ping 225.2.2.2 | Pings an IP multicast group address. A successful response indicates that the group address is functioning. |

Monitoring and Troubleshooting PIM

Monitoring PIM Information

Use the privileged EXEC commands in the following table to monitor your PIM configurations.

Table 15: PIM Monitoring Commands

| Command | Purpose |
|---|--|
| show ip pim all-vrfs tunnel [tunnel <i>tunnel_number</i> verbose] | Displays all VRFs. |
| show ip pim autorp | Displays global auto-RP information. |
| show ip pim boundary | Displays information about mroutes filtered by administratively scoped IPv4 multicast boundaries configured on an interface. |
| show ip pim interface | Displays information about interfaces configured for Protocol Independent Multicast (PIM). |
| show ip pim neighbor | Displays the PIM neighbor information. |
| show ip pim rp [<i>group-name</i> <i>group-address</i>] | Displays RP routers associated with a sparse-mode multicast group. This command is available in all software images. |
| show ip pim tunnel [tunnel verbose] | Displays information about Protocol Independent Multicast (PIM) tunnel interfaces |
| show ip pim vrf { word { all-vrfs autorp boundary bsr-router interface mdt neighbor rp rp-hash tunnel } } | Displays the VPN routing/forwarding instance. |
| show ip igmp groups detail | Displays the interested clients that have joined the specific multicast source group. |

Monitoring the RP Mapping and BSR Information

Use the privileged EXEC mode in the following table to verify the consistency of group-to-RP mappings:

Table 16: RP Mapping Monitoring Commands

| Command | Purpose |
|--|--|
| show ip pim rp [<i>hostname</i> or <i>IP address</i> mapping [<i>hostname</i> or <i>IP address</i> elected in-use] metric [<i>hostname</i> or <i>IP address</i>]] | <p>Displays all available RP mappings and metrics. This tells you how the device learns of the RP (through the BSR or the Auto-RP mechanism).</p> <ul style="list-style-type: none"> • (Optional) For the <i>hostname</i>, specify the IP name of the group about which to display RPs. • (Optional) For the <i>IP address</i>, specify the IP address of the group about which to display RPs. • (Optional) Use the mapping keyword to display all group-to-RP mappings of which the Cisco device is aware (either configured or learned from Auto-RP). • (Optional) Use the metric keyword to display the RP RPF metric. |
| show ip pim rp-hash <i>group</i> | <p>Displays the RP that was selected for the specified group. That is, on a PIMv2 router or multilayer switch, confirms that the same RP is the one that a PIMv1 system chooses. For <i>group</i>, enter the group address for which to display RP information.</p> |

Use the privileged EXEC commands in the following table to monitor BSR information:

Table 17: BSR Monitoring Commands

| Command | Purpose |
|-------------------------------|---|
| show ip pim bsr | Displays information about the elected BSR. |
| show ip pim bsr-router | Displays information about the BSRv2. |

Troubleshooting PIMv1 and PIMv2 Interoperability Problems

When debugging interoperability problems between PIMv1 and PIMv2, check these in the order shown:

1. Verify RP mapping with the **show ip pim rp-hash** privileged EXEC command, making sure that all systems agree on the same RP for the same group.
2. Verify interoperability between different versions of DRs and RPs. Make sure that the RPs are interacting with the DRs properly (by responding with register-stops and forwarding decapsulated data packets from registers).

Configuration Examples for PIM

Example: Enabling PIM Stub Routing

In this example, IP multicast routing is enabled, Switch A PIM uplink port 25 is configured as a routed uplink port with **spare-dense-mode** enabled. PIM stub routing is enabled on the VLAN 100 interfaces and on Gigabit Ethernet port 20.

```
Device(config)# ip multicast-routing
Device(config)# interface GigabitEthernet3/0/25
Device(config-if)# no switchport
Device(config-if)# ip address 3.1.1.2 255.255.255.0
Device(config-if)# ip pim sparse-dense-mode
Device(config-if)# exit
Device(config)# interface vlan100
Device(config-if)# ip pim passive
Device(config-if)# exit
Device(config)# interface GigabitEthernet3/0/20
Device(config-if)# ip pim passive
Device(config-if)# exit
Device(config)# interface vlan100
Device(config-if)# ip address 100.1.1.1 255.255.255.0
Device(config-if)# ip pim passive
Device(config-if)# exit
Device(config)# interface GigabitEthernet3/0/20
Device(config-if)# no switchport
Device(config-if)# ip address 10.1.1.1 255.255.255.0
Device(config-if)# ip pim passive
Device(config-if)# end
```

Example: Verifying PIM Stub Routing

To verify that PIM stub is enabled for each interface, use the **show ip pim interface** command in privileged EXEC mode:

```
Device# show ip pim interface
Address Interface Ver/ Nbr Query DR DR
Mode Count Intvl Prior
3.1.1.2 GigabitEthernet3/0/25 v2/SD 1 30 1 3.1.1.2
100.1.1.1 Vlan100 v2/P 0 30 1 100.1.1.1
10.1.1.1 GigabitEthernet3/0/20 v2/P 0 30 1 10.1.1.1
```

Example: Manually Assigning an RP to Multicast Groups

This example shows how to configure the address of the RP to 147.106.6.22 for multicast group 225.2.2.2 only:

```
Device(config)# access-list 1 permit 225.2.2.2 0.0.0.0
Device(config)# ip pim rp-address 147.106.6.22 1
```

Example: Configuring Auto-RP

This example shows how to send RP announcements out all PIM-enabled interfaces for a maximum of 31 hops. The IP address of port 1 is the RP. Access list 5 describes the group for which this device serves as RP:

```
Device(config)# ip pim send-rp-announce gigabitethernet1/0/1 scope 31 group-list 5
Device(config)# access-list 5 permit 224.0.0.0 15.255.255.255
```

Example: Sparse Mode with Auto-RP

The following example configures sparse mode with Auto-RP:

```
Device(config)# ip multicast-routing
Device(config)# ip pim autorp listener
Device(config)# ip pim send-rp-announce Loopback0 scope 16 group-list 1
Device(config)# ip pim send-rp-discovery Loopback1 scope 16
Device(config)# no ip pim dm-fallback
Device(config)# access-list 1 permit 239.254.2.0 0.0.0.255
Device(config)# access-list 1 permit 239.254.3.0 0.0.0.255
.
.
.
Device(config)# access-list 10 permit 224.0.1.39
Device(config)# access-list 10 permit 224.0.1.40
Device(config)# access-list 10 permit 239.254.2.0 0.0.0.255
Device(config)# access-list 10 permit 239.254.3.0 0.0.0.255
```

Example: Defining the IP Multicast Boundary to Deny Auto-RP Information

This example shows a portion of an IP multicast boundary configuration that denies Auto-RP information:

```
Device(config)# access-list 1 deny 224.0.1.39
Device(config)# access-list 1 deny 224.0.1.40
Device(config)# access-list 1 permit all
Device(config)# interface gigabitethernet1/0/1
Device(config-if)# ip multicast boundary 1
```

Example: Filtering Incoming RP Announcement Messages

This example shows a sample configuration on an Auto-RP mapping agent that is used to prevent candidate RP announcements from being accepted from unauthorized candidate RPs:

```
Device(config)# ip pim rp-announce-filter rp-list 10 group-list 20
Device(config)# access-list 10 permit host 172.16.5.1
Device(config)# access-list 10 permit host 172.16.2.1
Device(config)# access-list 20 deny 239.0.0.0 0.0.255.255
Device(config)# access-list 20 permit 224.0.0.0 15.255.255.255
```

The mapping agent accepts candidate RP announcements from only two devices, 172.16.5.1 and 172.16.2.1. The mapping agent accepts candidate RP announcements from these two devices only for multicast groups that fall in the group range of 224.0.0.0 to 239.255.255.255. The mapping agent does not accept candidate

RP announcements from any other devices in the network. Furthermore, the mapping agent does not accept candidate RP announcements from 172.16.5.1 or 172.16.2.1 if the announcements are for any groups in the 239.0.0.0 through 239.255.255.255 range. This range is the administratively scoped address range.

Example: Preventing Join Messages to False RPs

If all interfaces are in sparse mode, use a default-configured RP to support the two well-known groups 224.0.1.39 and 224.0.1.40. Auto-RP uses these two well-known groups to collect and distribute RP-mapping information. When this is the case and the **ip pim accept-rp auto-rp** command is configured, another **ip pim accept-rp** command accepting the RP must be configured as follows:

```
Device(config)# ip pim accept-rp 172.10.20.1 1
Device(config)# access-list 1 permit 224.0.1.39
Device(config)# access-list 1 permit 224.0.1.40
```

Example: Configuring Candidate BSRs

This example shows how to configure a candidate BSR, which uses the IP address 172.21.24.18 on a port as the advertised BSR address, uses 30 bits as the hash-mask-length, and has a priority of 10.

```
Device(config)# interface gigabitethernet1/0/2
Device(config-if)# ip address 172.21.24.18 255.255.255.0
Device(config-if)# ip pim sparse-mode
Device(config-if)# ip pim bsr-candidate gigabitethernet1/0/2 30 10
```

Example: Configuring Candidate RPs

This example shows how to configure the device to advertise itself as a candidate RP to the BSR in its PIM domain. Standard access list number 4 specifies the group prefix associated with the RP that has the address identified by a port. That RP is responsible for the groups with the prefix 239.

```
Device(config)# ip pim rp-candidate gigabitethernet1/0/2 group-list 4
Device(config)# access-list 4 permit 239.0.0.0 0.255.255.255
```

Feature History for PIM

This table provides release and related information for the features explained in this module.

These features are available in all the releases subsequent to the one they were introduced in, unless noted otherwise.

| Release | Feature | Feature Information |
|--------------------------------|--------------------------|--|
| Cisco IOS XE Fuji 16.9.2 | PIM | The Protocol Independent Multicast (PIM) protocol maintains the current IP multicast service mode of receiver-initiated membership. PIM is not dependent on a specific unicast routing protocol; it is IP routing protocol independent and can leverage whichever unicast routing protocols are used to populate the unicast routing table, including Enhanced Interior Gateway Routing Protocol (EIGRP), Open Shortest Path First (OSPF), and static routes. PIM uses unicast routing information to perform the multicast forwarding function. |
| Cisco IOS XE Gibraltar 16.12.1 | Bidirectional PIM | Bidirectional PIM is a variant of the PIM suite of routing protocols for IP multicast. |
| Cisco IOS XE Cupertino 17.7.1 | PIM Bidirectional PIM | These features were implemented on supervisor modules C9400X-SUP-2 and C9400X-SUP-2XL, which were introduced in this release. |
| Cisco IOS XE Cupertino 17.9.1 | PIM Bidirectional PIM | This feature was implemented on C9200CX-12P-2X2G, C9200CX-8P-2X2G, and C9200CX-12T-2X2G models of the Cisco Catalyst 9200CX Series Switches, which were introduced in this release. |

Use the Cisco Feature Navigator to find information about platform and software image support. To access Cisco Feature Navigator, go to <http://www.cisco.com/go/cfn>.



CHAPTER 8

Configuring PIM MIB Extension for IP Multicast

- [Information About PIM MIB Extension for IP Multicast, on page 175](#)
- [How to Configure PIM MIB Extension for IP Multicast, on page 176](#)
- [Configuration Examples for PIM MIB Extensions, on page 177](#)
- [Additional References for PIM MIB Extension for IP Multicast, on page 178](#)
- [Feature History for PIM MIB Extension for IP Multicast, on page 178](#)

Information About PIM MIB Extension for IP Multicast

PIM MIB Extensions for SNMP Traps for IP Multicast

Protocol Independent Multicast (PIM) is an IP multicast routing protocol used for routing multicast data packets to multicast groups. RFC 2934 defines the PIM MIB for IPv4, which describes managed objects that enable users to remotely monitor and configure PIM using Simple Network Management Protocol (SNMP).

PIM MIB extensions introduce the following new classes of PIM notifications:

- neighbor-change--This notification results from the following conditions:
 - A router's PIM interface is disabled or enabled (using the **ip pim** command in interface configuration mode)
 - A router's PIM neighbor adjacency expires (defined in RFC 2934)
- rp-mapping-change--This notification results from a change in the rendezvous point (RP) mapping information due to either Auto-RP messages or bootstrap router (BSR) messages.
- invalid-pim-message--This notification results from the following conditions:
 - An invalid (*, G) Join or Prune message is received by the device (for example, when a router receives a Join or Prune message for which the RP specified in the packet is not the RP for the multicast group)
 - An invalid PIM register message is received by the device (for example, when a router receives a register message from a multicast group for which it is not the RP)

Benefits of PIM MIB Extensions

PIM MIB extensions:

- Allow users to identify changes in the multicast topology of their network by detecting changes in the RP mapping.
- Provide traps to monitor the PIM protocol on PIM-enabled interfaces.
- Help users identify routing issues when multicast neighbor adjacencies expire on a multicast interface.
- Enable users to monitor RP configuration errors (for example, errors due to flapping in dynamic RP allocation protocols like Auto-RP).

How to Configure PIM MIB Extension for IP Multicast

Enabling PIM MIB Extensions for IP Multicast

Perform this task to enable PIM MIB extensions for IP multicast.



Note

- The `pimInterfaceVersion` object was removed from RFC 2934 and, therefore, is no longer supported in software.
- The following MIB tables are not supported in Cisco software:
 - `pimIpMRouteTable`
 - `pimIpMRouteNextHopTable`

Procedure

| | Command or Action | Purpose |
|---------------|--|---|
| Step 1 | enable Example: Device> enable | Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted. |
| Step 2 | configure terminal Example: Device# configure terminal | Enters global configuration mode. |
| Step 3 | snmp-server enable traps pim [neighbor-change rp-mapping-change invalid-pim-message] Example: | Enables a device to send PIM notifications. <ul style="list-style-type: none"> • neighbor-change --This keyword enables notifications indicating when a device's PIM interface is disabled or enabled, or |

| | Command or Action | Purpose |
|---------------|---|---|
| | <pre>Device(config)# snmp-server enable traps pim neighbor-change</pre> | <p>when a device's PIM neighbor adjacency expires.</p> <ul style="list-style-type: none"> • rp-mapping-change --This keyword enables notifications indicating a change in RP mapping information due to either Auto-RP messages or BSR messages. • invalid-pim-message --This keyword enables notifications for monitoring invalid PIM protocol operations (for example, when a device receives a join or prune message for which the RP specified in the packet is not the RP for the multicast group or when a device receives a register message from a multicast group for which it is not the RP). |
| Step 4 | <p>snmp-server host <i>host-address</i> [traps informs] <i>community-string</i> pim</p> <p>Example:</p> <pre>Device(config)# snmp-server host 10.10.10.10 traps public pim</pre> | Specifies the recipient of a PIM SNMP notification operation. |

Configuration Examples for PIM MIB Extensions

Example Enabling PIM MIB Extensions for IP Multicast

The following example shows how to configure a router to generate notifications indicating that a PIM interface of the router has been enabled. The first line configures PIM traps to be sent as SNMP v2c traps to the host with IP address 10.0.0.1. The second line configures the router to send the neighbor-change class of trap notification to the host.

```
snmp-server host 10.0.0.1 traps version 2c public pim
snmp-server enable traps pim neighbor-change
interface ethernet0/0
 ip pim sparse-mode
```

Additional References for PIM MIB Extension for IP Multicast

Related Documents

| Related Topic | Document Title |
|--|---|
| For complete syntax and usage information for the commands used in this chapter. | See the IP Multicast Routing Commands section of the <i>Command Reference (Catalyst 9200 Series Switches)</i> |

Standards and RFCs

| Standard/RFC | Title |
|-------------------------------------|--|
| draft-kouvelas-pim-bidir-new-00.txt | A New Proposal for Bi-directional PIM |
| RFC 1112 | Host Extensions for IP Multicasting |
| RFC 1918 | Address Allocation for Private Internets |
| RFC 2770 | GLOP Addressing in 233/8 |
| RFC 3569 | An Overview of Source-Specific Multicast (SSM) |

Feature History for PIM MIB Extension for IP Multicast

This table provides release and related information for the features explained in this module.

These features are available in all the releases subsequent to the one they were introduced in, unless noted otherwise

| Release | Feature | Feature Information |
|-------------------------------|------------------------------------|---|
| Cisco IOS XE Fuji 16.9.2 | PIM MIB Extension for IP Multicast | Protocol Independent Multicast (PIM) is an IP multicast routing protocol used for routing multicast data packets to multicast groups. RFC 2934 defines the PIM MIB for IPv4, which describes managed objects that enable users to remotely monitor and configure PIM using Simple Network Management Protocol (SNMP). |
| Cisco IOS XE Cupertino 17.7.1 | PIM MIB Extension for IP Multicast | This feature was implemented on supervisor modules C9400X-SUP-2 and C9400X-SUP-2XL, which were introduced in this release. |
| Cisco IOS XE Cupertino 17.9.1 | PIM MIB Extension for IP Multicast | This feature was implemented on C9200CX-12P-2X2G, C9200CX-8P-2X2G, and C9200CX-12T-2X2G models of the Cisco Catalyst 9200CX Series Switches, which were introduced in this release. |

Use the Cisco Feature Navigator to find information about platform and software image support. To access Cisco Feature Navigator, go to <http://www.cisco.com/go/cfn>.



CHAPTER 9

Configuring SSM

- [Prerequisites for Configuring SSM, on page 181](#)
- [Restrictions for Configuring SSM, on page 181](#)
- [Information About SSM, on page 183](#)
- [How to Configure SSM, on page 186](#)
- [Monitoring SSM, on page 192](#)
- [Where to Go Next for SSM, on page 192](#)
- [Additional References for SSM, on page 193](#)
- [Feature History for SSM, on page 193](#)

Prerequisites for Configuring SSM

The following are the prerequisites for configuring source-specific multicast (SSM) and SSM mapping:

- Before you configure SSM mapping, you must perform the following tasks:
 - Enable IP multicast routing.
 - Enable PIM sparse mode.
 - Configure SSM.
- Before you configure static SSM mapping, you must configure access control lists (ACLs) that define the group ranges to be mapped to source addresses.
- Before you can configure and use SSM mapping with DNS lookups, you need to add records to a running DNS server. If you do not already have a DNS server running, you need to install one.



Note You can use a product such as *Cisco Network Registrar* to add records to a running DNS server.

Restrictions for Configuring SSM

The following are the restrictions for configuring SSM:

- To run SSM with IGMPv3, SSM must be supported in the Cisco IOS router, the host where the application is running, and the application itself.
- Existing applications in a network predating SSM will not work within the SSM range unless they are modified to support (S, G) channel subscriptions. Therefore, enabling SSM in a network may cause problems for existing applications if they use addresses within the designated SSM range.
- IGMP Snooping—IGMPv3 uses new membership report messages that might not be correctly recognized by older IGMP snooping devices.
- Address management is still necessary to some degree when SSM is used with Layer 2 switching mechanisms. Cisco Group Management Protocol (CGMP), IGMP snooping, or Router-Port Group Management Protocol (RGMP) support only group-specific filtering, not (S, G) channel-specific filtering. If different receivers in a switched network request different (S, G) channels sharing the same group, they do not benefit from these existing mechanisms. Instead, both receivers receive all (S, G) channel traffic and filter out the unwanted traffic on input. Because SSM can re-use the group addresses in the SSM range for many independent applications, this situation can lead to decreased traffic filtering in a switched network. For this reason, it is important to use random IP addresses from the SSM range for an application to minimize the chance for re-use of a single address within the SSM range between different applications. For example, an application service providing a set of television channels should, even with SSM, use a different group for each television (S, G) channel. This setup guarantees that multiple receivers to different channels within the same application service never experience traffic aliasing in networks that include Layer 2 devices.
- In PIM-SSM, the last hop router will continue to periodically send (S, G) join messages if appropriate (S, G) subscriptions are on the interfaces. Therefore, as long as receivers send (S, G) subscriptions, the shortest path tree (SPT) state from the receivers to the source will be maintained, even if the source is not sending traffic for longer periods of time (or even never).

The opposite situation occurs with PIM-SM, where (S, G) state is maintained only if the source is sending traffic and receivers are joining the group. If a source stops sending traffic for more than 3 minutes in PIM-SM, the (S, G) state is deleted and only reestablished after packets from the source arrive again through the RPT (rendezvous point tree). Because no mechanism in PIM-SSM notifies a receiver that a source is active, the network must maintain the (S, G) state in PIM-SSM as long as receivers are requesting receipt of that channel.

The following are the restrictions for configuring SSM mapping:

- The SSM Mapping feature does not share the benefit of full SSM. SSM mapping takes a group G join from a host and identifies this group with an application associated with one or more sources, therefore, it can only support one such application per group G. Nevertheless, full SSM applications may still share the same group also used in SSM mapping.
- Enable IGMPv3 with care on the last hop router when you rely solely on SSM mapping as a transition solution for full SSM. When you enable both SSM mapping and IGMPv3 and the hosts already support IGMPv3 (but not SSM), the hosts send IGMPv3 group reports. SSM mapping does not support these IGMPv3 group reports, and the router does not correctly associate sources with these reports.

Information About SSM

The source-specific multicast (SSM) feature is an extension of IP multicast in which datagram traffic is forwarded to receivers from only those multicast sources that the receivers have explicitly joined. For multicast groups configured for SSM, only SSM distribution trees (no shared trees) are created.

This section describes how to configure source-specific multicast (SSM). For a complete description of the SSM commands in this section, refer to the *IP Multicast Command Reference*.

SSM Components Overview

SSM is a datagram delivery model that best supports one-to-many applications, also known as broadcast applications. SSM is a core networking technology for the Cisco implementation of IP multicast solutions targeted for audio and video broadcast application environments. The device supports the following components that support SSM implementation:

- Protocol independent multicast source-specific mode (PIM-SSM)

PIM-SSM is the routing protocol that supports the implementation of SSM and is derived from PIM sparse mode (PIM-SM).

- Internet Group Management Protocol version 3 (IGMPv3)

SSM and Internet Standard Multicast (ISM)

The current IP multicast infrastructure in the Internet and many enterprise intranets is based on the PIM-SM protocol and Multicast Source Discovery Protocol (MSDP). These protocols have the limitations of the Internet Standard Multicast (ISM) service model. For example, with ISM, the network must maintain knowledge about which hosts in the network are actively sending multicast traffic.

The ISM service consists of the delivery of IP datagrams from any source to a group of receivers called the multicast host group. The datagram traffic for the multicast host group consists of datagrams with an arbitrary IP unicast source address (S) and the multicast group address (G) as the IP destination address. Systems receive this traffic by becoming members of the host group. Membership in a host group simply requires signaling the host group through IGMP version 1, 2, or 3.

In SSM, delivery of datagrams is based on (S, G) channels. In both SSM and ISM, no signaling is required to become a source. However, in SSM, receivers must subscribe or unsubscribe to (S, G) channels to receive or not receive traffic from specific sources. In other words, receivers can receive traffic only from (S, G) channels to which they are subscribed, whereas in ISM, receivers need not know the IP addresses of sources from which they receive their traffic. The proposed standard approach for channel subscription signaling uses IGMP and includes modes membership reports, which are supported only in IGMP version 3.

SSM IP Address Range

SSM can coexist with the ISM service by applying the SSM delivery model to a configured subset of the IP multicast group address range. Cisco IOS software allows SSM configuration for the IP multicast address range of 224.0.0.0 through 239.255.255.255. When an SSM range is defined, existing IP multicast receiver applications do not receive any traffic when they try to use an address in the SSM range (unless the application is modified to use an explicit (S, G) channel subscription).

SSM Operations

An established network, in which IP multicast service is based on PIM-SM, can support SSM services. SSM can also be deployed alone in a network without the full range of protocols required for interdomain PIM-SM (for example, MSDP, Auto-RP, or bootstrap router [BSR]) if only SSM service is needed.

If SSM is deployed in a network already configured for PIM-SM, only the last-hop routers support SSM. Routers that are not directly connected to receivers do not require support for SSM. In general, these not-last-hop routers must only run PIM-SM in the SSM range and might need additional access control configuration to suppress MSDP signalling, registering, or PIM-SM shared tree operations from occurring within the SSM range.

Use the **`ip pim ssm`** global configuration command to configure the SSM range and to enable SSM. This configuration has the following effects:

- For groups within the SSM range, (S, G) channel subscriptions are accepted through IGMPv3 include-mode membership reports.
- PIM operations within the SSM range of addresses change to PIM-SSM, a mode derived from PIM-SM. In this mode, only PIM (S, G) join and prune messages are generated by the router, and no (S, G) rendezvous point tree (RPT) or (*, G) RPT messages are generated. Incoming messages related to RPT operations are ignored or rejected, and incoming PIM register messages are immediately answered with register-stop messages. PIM-SSM is backward-compatible with PIM-SM unless a router is a last-hop router. Therefore, routers that are not last-hop routers can run PIM-SM for SSM groups (for example, if they do not yet support SSM).
- No MSDP source-active (SA) messages within the SSM range are accepted, generated, or forwarded.

SSM Mapping

In a typical set-top box (STB) deployment, each TV channel uses one separate IP multicast group and has one active server host sending the TV channel. A single server can send multiple TV channels, but each to a different group. In this network environment, if a router receives an IGMPv1 or IGMPv2 membership report for a particular group, the report addresses the well-known TV server for the TV channel associated with the multicast group.

When SSM mapping is configured, if a router receives an IGMPv1 or IGMPv2 membership report for a particular group, the router translates this report into one or more channel memberships for the well-known sources associated with this group.

When the router receives an IGMPv1 or IGMPv2 membership report for a group, the router uses SSM mapping to determine one or more source IP addresses for the group. SSM mapping then translates the membership report as an IGMPv3 report and continues as if it had received an IGMPv3 report. The router then sends PIM joins and continues to be joined to these groups as long as it continues to receive the IGMPv1 or IGMPv2 membership reports, and the SSM mapping for the group remains the same.

SSM mapping enables the last hop router to determine the source addresses either by a statically configured table on the router or through a DNS server. When the statically configured table or the DNS mapping changes, the router leaves the current sources associated with the joined groups.

Static SSM Mapping

With static SSM mapping, you can configure the last hop router to use a static map to determine the sources that are sending to groups. Static SSM mapping requires that you configure ACLs to define group ranges.

After configuring the ACLs to define group ranges, you can then map the groups permitted by those ACLs to sources by using the **ip igmp ssm-map static** global configuration command.

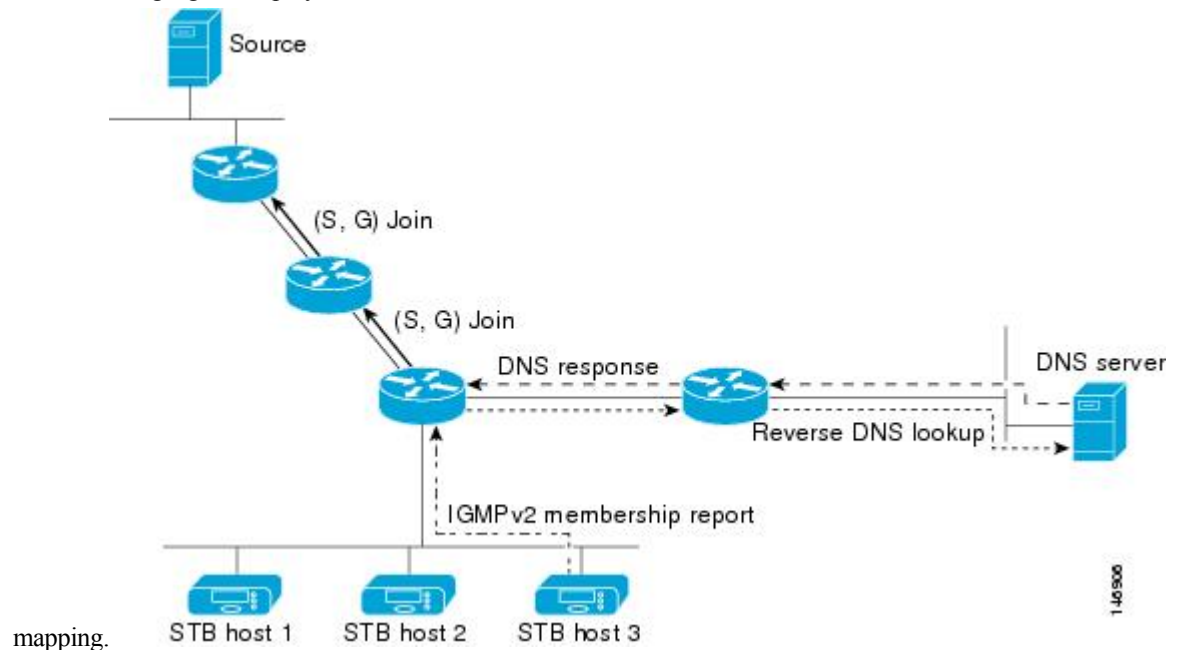
You can configure static SSM mapping in smaller networks when a DNS is not needed or to locally override DNS mappings. When configured, static SSM mappings take precedence over DNS mappings.

DNS-Based SSM Mapping

You can use DNS-based SSM mapping to configure the last hop router to perform a reverse DNS lookup to determine sources sending to groups. When DNS-based SSM mapping is configured, the router constructs a domain name that includes the group address and performs a reverse lookup into the DNS. The router looks up IP address resource records and uses them as the source addresses associated with this group. SSM mapping supports up to 20 sources for each group. The router joins all sources configured for a group.

Figure 16: DNS-Based SSM Mapping

The following figure displays DNS-based SSM



The SSM mapping mechanism that enables the last hop router to join multiple sources for a group can provide source redundancy for a TV broadcast. In this context, the last hop router provides redundancy using SSM mapping to simultaneously join two video sources for the same TV channel. However, to prevent the last hop router from duplicating the video traffic, the video sources must use a server-side switchover mechanism. One video source is active, and the other backup video source is passive. The passive source waits until an active source failure is detected before sending the video traffic for the TV channel. Thus, the server-side switchover mechanism ensures that only one of the servers is actively sending video traffic for the TV channel.

To look up one or more source addresses for a group that includes G1, G2, G3, and G4, you must configure these DNS records on the DNS server:

```
G4.G3.G2.G1 [multicast-domain] [timeout] IN A source-address-1
IN A source-address-2
IN A source-address-n
```

See your DNS server documentation for more information about configuring DNS resource records.

How to Configure SSM

Configuring SSM

Follow these steps to configure SSM:

This procedure is optional.

Before you begin

If you want to use an access list to define the Source Specific Multicast (SSM) range, configure the access list before you reference the access list in the **ip pim ssm** command.

Procedure

| | Command or Action | Purpose |
|---------------|---|--|
| Step 1 | enable Example: Device> enable | Enables privileged EXEC mode. • Enter your password if prompted. |
| Step 2 | configure terminal Example: Device# configure terminal | Enters global configuration mode. |
| Step 3 | ip pim ssm [default range <i>access-list</i>] Example: Device(config)# ip pim ssm range 20 | Defines the SSM range of IP multicast addresses. |
| Step 4 | interface <i>type number</i> Example: Device(config)# interface gigabitethernet 1/0/1 | Selects an interface that is connected to hosts on which IGMPv3 can be enabled, and enters the interface configuration mode. |
| Step 5 | ip pim {sparse-mode } Example: Device(config-if)# ip pim sparse-mode | Enables PIM on an interface. |

| | Command or Action | Purpose |
|---------------|---|--|
| Step 6 | ip igmp version 3 Example: <pre>Device(config-if)# ip igmp version 3</pre> | Enables IGMPv3 on this interface. The default version of IGMP is set to Version 2. |
| Step 7 | end Example: <pre>Device(config)# end</pre> | Returns to privileged EXEC mode. |
| Step 8 | show running-config Example: <pre>Device# show running-config</pre> | Verifies your entries. |
| Step 9 | copy running-config startup-config Example: <pre>Device# copy running-config startup-config</pre> | (Optional) Saves your entries in the configuration file. |

Configuring Source Specific Multicast Mapping

The Source Specific Multicast (SSM) mapping feature supports SSM transition when supporting SSM on the end system is impossible or unwanted due to administrative or technical reasons. You can use SSM mapping to leverage SSM for video delivery to legacy STBs that do not support IGMPv3 or for applications that do not use the IGMPv3 host stack.

Configuring Static SSM Mapping

Follow these steps to configure static SSM Mapping:

Procedure

| | Command or Action | Purpose |
|---------------|--|--|
| Step 1 | enable Example: <pre>Device> enable</pre> | Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted. |
| Step 2 | configure terminal Example: | Enters global configuration mode. |

| | Command or Action | Purpose |
|---------------|---|--|
| | Device# <code>configure terminal</code> | |
| Step 3 | <p>ip igmp ssm-map enable</p> <p>Example:</p> <pre>Device(config)# ip igmp ssm-map enable</pre> | <p>Enables SSM mapping for groups in the configured SSM range.</p> <p>Note By default, this command enables DNS-based SSM mapping.</p> |
| Step 4 | <p>no ip igmp ssm-map query dns</p> <p>Example:</p> <pre>Device(config)# no ip igmp ssm-map query dns</pre> | <p>(Optional) Disables DNS-based SSM mapping.</p> <p>Note Disable DNS-based SSM mapping if you only want to rely on static SSM mapping. By default, the ip igmp ssm-map command enables DNS-based SSM mapping.</p> |
| Step 5 | <p>ip igmp ssm-map static <i>access-list</i> <i>source-address</i></p> <p>Example:</p> <pre>Device(config)# ip igmp ssm-map static 11 172.16.8.11</pre> | <p>Configures static SSM mapping.</p> <ul style="list-style-type: none"> The ACL supplied for the <i>access-list</i> argument defines the groups to be mapped to the source IP address entered for the <i>source-address</i> argument. <p>Note You can configure additional static SSM mappings. If additional SSM mappings are configured and the router receives an IGMPv1 or IGMPv2 membership report for a group in the SSM range, the device determines the source addresses associated with the group by walking each configured ip igmp ssm-map static command. The device associates up to 20 sources per group.</p> <p>Repeat Step to configure additional static SSM mappings, if required.</p> |
| Step 6 | <p>end</p> <p>Example:</p> <pre>Device(config)# end</pre> | <p>Returns to privileged EXEC mode.</p> |
| Step 7 | <p>show running-config</p> <p>Example:</p> | <p>Verifies your entries.</p> |

| | Command or Action | Purpose |
|---------------|---|--|
| | Device# <code>show running-config</code> | |
| Step 8 | copy running-config startup-config Example: Device# <code>copy running-config startup-config</code> | (Optional) Saves your entries in the configuration file. |

Configuring DNS-Based SSM Mapping

To configure DNS-based SSM mapping, you need to create a DNS server zone or add records to an existing zone. If the routers that are using DNS-based SSM mapping are also using DNS for other purposes, you should use a normally configured DNS server. If DNS-based SSM mapping is the only DNS implementation being used on the router, you can configure a false DNS setup with an empty root zone or a root zone that points back to itself.

Procedure

| | Command or Action | Purpose |
|---------------|---|--|
| Step 1 | enable Example: Device> <code>enable</code> | Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted. |
| Step 2 | configure terminal Example: Device# <code>configure terminal</code> | Enters global configuration mode. |
| Step 3 | ip igmp ssm-map enable Example: Device(config)# <code>ip igmp ssm-map enable</code> | Enables SSM mapping for groups in a configured SSM range. |
| Step 4 | ip igmp ssm-map query dns Example: Device(config)# <code>ip igmp ssm-map query dns</code> | (Optional) Enables DNS-based SSM mapping. <ul style="list-style-type: none"> • By default, the ip igmp ssm-map command enables DNS-based SSM mapping. Only the no form of this command is saved to the running configuration. |

| | Command or Action | Purpose |
|----------------|--|---|
| | | Note Use this command to reenable DNS-based SSM mapping if DNS-based SSM mapping is disabled. |
| Step 5 | ip domain multicast <i>domain-prefix</i> Example: Device(config)# ip domain multicast ssm-map.cisco.com | (Optional) Changes the domain prefix used for DNS-based SSM mapping. • By default, the software uses the ip-addr.arpa domain prefix. |
| Step 6 | ip name-server <i>server-address1</i> [<i>server-address2...server-address6</i>] Example: Device(config)# ip name-server 10.48.81.21 | Specifies the address of one or more name servers to use for name and address resolution. |
| Step 7 | Repeat Step 6 to configure additional DNS servers for redundancy, if required. | |
| Step 8 | end Example: Device(config)# end | Returns to privileged EXEC mode. |
| Step 9 | show running-config Example: Device# show running-config | Verifies your entries. |
| Step 10 | copy running-config startup-config Example: Device# copy running-config startup-config | (Optional) Saves your entries in the configuration file. |

Configuring Static Traffic Forwarding with SSM Mapping

Follow these steps to configure static traffic forwarding with SSM mapping on the last hop router:

Procedure

| | Command or Action | Purpose |
|---------------|----------------------------------|---|
| Step 1 | enable Example: | Enables privileged EXEC mode. • Enter your password if prompted. |

| | Command or Action | Purpose |
|---------------|---|--|
| | Device> enable | |
| Step 2 | configure terminal Example: Device# configure terminal | Enters global configuration mode. |
| Step 3 | interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet 1/0/1 | Selects an interface on which to statically forward traffic for a multicast group using SSM mapping, and enters interface configuration mode. Note Static forwarding of traffic with SSM mapping works with either DNS-based SSM mapping or statically configured SSM mapping. |
| Step 4 | ip igmp static-group <i>group-address</i> source <i>source</i> ssm-map Example: Device(config-if)# ip igmp static-group 239.1.2.1 source ssm-map | Configures SSM mapping to statically forward a (S, G) channel from the interface. Use this command if you want to statically forward SSM traffic for certain groups. Use DNS-based SSM mapping to determine the source addresses of the channels. |
| Step 5 | end Example: Device(config)# end | Returns to privileged EXEC mode. |
| Step 6 | show running-config Example: Device# show running-config | Verifies your entries. |
| Step 7 | copy running-config startup-config Example: Device# copy running-config startup-config | (Optional) Saves your entries in the configuration file. |

Monitoring SSM

Use the privileged EXEC commands in the following table to monitor SSM.

Table 18: Commands for Monitoring SSM

| Command | Purpose |
|-----------------------------------|--|
| show ip igmp groups detail | Displays the (S, G) channel subscription through IGMPv3. |
| show ip mroute | Displays whether a multicast group supports SSM service or whether a source-specific host report was received. |

Monitoring SSM Mapping

Use the privileged EXEC commands in the following table to monitor SSM mapping.

Table 19: SSM Mapping Monitoring Commands

| Command | Purpose |
|--|--|
| show ip igmp ssm-mapping | Displays information about SSM mapping. |
| show ip igmp ssm-mapping group-address | Displays the sources that SSM mapping uses for group. |
| show ip igmp groups [<i>group-name</i> <i>group-address</i> <i>interface-type interface-number</i>] [detail] | Displays the multicast groups with receivers that are connected to the router and that were learned through SSM mapping. |
| show host | Displays the default domain name, the style of name resolution service, a list of name server hosts, and the cache of hostnames and addresses. |
| debug ip igmp group-address | Displays the IGMP packets received and sent and other host-related events. |

Where to Go Next for SSM

You can configure the following:

- IGMP
- PIM
- IP Multicast Routing
- Service Discovery Gateway

Additional References for SSM

Related Documents

| Related Topic | Document Title |
|--|---|
| For complete syntax and usage information for the commands used in this chapter. | See the IP Multicast Routing Commands section of the <i>Command Reference (Catalyst 9200 Series Switches)</i> |

Standards and RFCs

| Standard/RFC | Title |
|--------------|--|
| RFC 4601 | <i>Protocol-Independent Multicast-Sparse Mode (PIM-SM): Protocol Specification</i> |

Feature History for SSM

This table provides release and related information for the features explained in this module.

These features are available in all the releases subsequent to the one they were introduced in, unless noted otherwise.

| Release | Feature | Feature Information |
|-------------------------------|---------|--|
| Cisco IOS XE Fuji 16.9.2 | SSM | SSM is an extension of IP multicast in which datagram traffic is forwarded to receivers from only those multicast sources that the receivers have explicitly joined. For multicast groups configured for SSM, only SSM distribution trees (no shared trees) are created. |
| Cisco IOS XE Cupertino 17.9.1 | SSM | This feature was implemented on C9200CX-12P-2X2G, C9200CX-8P-2X2G, and C9200CX-12T-2X2G models of the Cisco Catalyst 9200CX Series Switches, which were introduced in this release. |

Use the Cisco Feature Navigator to find information about platform and software image support. To access Cisco Feature Navigator, go to <http://www.cisco.com/go/cfn>.



CHAPTER 10

Implementing IPv6 Multicast

- [Information About Implementing IPv6 Multicast Routing, on page 195](#)
- [How to Implement IPv6 Multicast, on page 203](#)
- [Additional References, on page 226](#)
- [Feature History for IPv6 Multicast, on page 226](#)

Information About Implementing IPv6 Multicast Routing

This chapter describes how to implement IPv6 multicast routing on the switch.

Traditional IP communication allows a host to send packets to a single host (unicast transmission) or to all hosts (broadcast transmission). IPv6 multicast provides a third scheme, allowing a host to send a single data stream to a subset of all hosts (group transmission) simultaneously.

IPv6 Multicast Overview

An IPv6 multicast group is an arbitrary group of receivers that want to receive a particular data stream. This group has no physical or geographical boundaries--receivers can be located anywhere on the Internet or in any private network. Receivers that are interested in receiving data flowing to a particular group must join the group by signaling their local switch. This signaling is achieved with the MLD protocol.

Switches use the MLD protocol to learn whether members of a group are present on their directly attached subnets. Hosts join multicast groups by sending MLD report messages. The network then delivers data to a potentially unlimited number of receivers, using only one copy of the multicast data on each subnet. IPv6 hosts that wish to receive the traffic are known as group members.

Packets delivered to group members are identified by a single multicast group address. Multicast packets are delivered to a group using best-effort reliability, just like IPv6 unicast packets.

The multicast environment consists of senders and receivers. Any host, regardless of whether it is a member of a group, can send to a group. However, only members of a group can listen to and receive the message.

A multicast address is chosen for the receivers in a multicast group. Senders use that address as the destination address of a datagram to reach all members of the group.



Note As per RFC 4291, the FF0x::/12 (where the T flag is set to 0 in IPv6 destination address) is for permanently assigned (“well-known”) IPv6 multicast address range.

In Cisco Catalyst 9200 Series Switches, the default behavior for packets with this address range is to flood in the ingress VLAN.

Membership in a multicast group is dynamic; hosts can join and leave at any time. There is no restriction on the location or number of members in a multicast group. A host can be a member of more than one multicast group at a time.

How active a multicast group is, its duration, and its membership can vary from group to group and from time to time. A group that has members may have no activity.

IPv6 Multicast Routing Implementation

The Cisco IOS software supports the following protocols to implement IPv6 multicast routing:

- MLD is used by IPv6 switches to discover multicast listeners (nodes that want to receive multicast packets destined for specific multicast addresses) on directly attached links. There are two versions of MLD: MLD version 1 is based on version 2 of the Internet Group Management Protocol (IGMP) for IPv4, and MLD version 2 is based on version 3 of the IGMP for IPv4. IPv6 multicast for Cisco IOS software uses both MLD version 2 and MLD version 1. MLD version 2 is fully backward-compatible with MLD version 1 (described in RFC 2710). Hosts that support only MLD version 1 will interoperate with a switch running MLD version 2. Mixed LANs with both MLD version 1 and MLD version 2 hosts are likewise supported.
- PIM-SM is used between switches so that they can track which multicast packets to forward to each other and to their directly connected LANs.
- PIM in Source Specific Multicast (PIM-SSM) is similar to PIM-SM with the additional ability to report interest in receiving packets from specific source addresses (or from all but the specific source addresses) to an IP multicast address.

IPv6 Multicast Listener Discovery Protocol

To start implementing multicasting in the campus network, users must first define who receives the multicast. The MLD protocol is used by IPv6 switches to discover the presence of multicast listeners (for example, nodes that want to receive multicast packets) on their directly attached links, and to discover specifically which multicast addresses are of interest to those neighboring nodes. It is used for discovering local group and source-specific group membership.

The MLD protocol provides a means to automatically control and limit the flow of multicast traffic throughout your network with the use of special multicast queriers and hosts.

Multicast Queriers and Hosts

A multicast querier is a network device, such as a switch, that sends query messages to discover which network devices are members of a given multicast group.

A multicast host is a receiver, including switches, that send report messages to inform the querier of a host membership.

A set of queriers and hosts that receive multicast data streams from the same source is called a multicast group. Queriers and hosts use MLD reports to join and leave multicast groups and to begin receiving group traffic.

MLD uses the Internet Control Message Protocol (ICMP) to carry its messages. All MLD messages are link-local with a hop limit of 1, and they all have the switch alert option set. The switch alert option implies an implementation of the hop-by-hop option header.

MLD Access Group

The MLD access group provides receiver access control in Cisco IOS IPv6 multicast switches. This feature limits the list of groups a receiver can join, and it allows or denies sources used to join SSM channels.

Explicit Tracking of Receivers

The explicit tracking feature allows a switch to track the behavior of the hosts within its IPv6 network. This feature also enables the fast leave mechanism to be used with MLD version 2 host reports.

Protocol Independent Multicast

Protocol Independent Multicast (PIM) is used between switches so that they can track which multicast packets to forward to each other and to their directly connected LANs. PIM works independently of the unicast routing protocol to perform send or receive multicast route updates like other protocols. Regardless of which unicast routing protocols are being used in the LAN to populate the unicast routing table, Cisco IOS PIM uses the existing unicast table content to perform the Reverse Path Forwarding (RPF) check instead of building and maintaining its own separate routing table.

You can configure IPv6 multicast to use either PIM-SM or PIM-SSM operation, or you can use both PIM-SM and PIM-SSM together in your network.

PIM-Sparse Mode

IPv6 multicast provides support for intradomain multicast routing using PIM-SM. PIM-SM uses unicast routing to provide reverse-path information for multicast tree building, but it is not dependent on any particular unicast routing protocol.

PIM-SM is used in a multicast network when relatively few switches are involved in each multicast and these switches do not forward multicast packets for a group, unless there is an explicit request for the traffic. PIM-SM distributes information about active sources by forwarding data packets on the shared tree. PIM-SM initially uses shared trees, which requires the use of an RP.

Requests are accomplished via PIM joins, which are sent hop by hop toward the root node of the tree. The root node of a tree in PIM-SM is the RP in the case of a shared tree or the first-hop switch that is directly connected to the multicast source in the case of a shortest path tree (SPT). The RP keeps track of multicast groups and the hosts that send multicast packets are registered with the RP by that host's first-hop switch.

As a PIM join travels up the tree, switches along the path set up multicast forwarding state so that the requested multicast traffic will be forwarded back down the tree. When multicast traffic is no longer needed, a switch sends a PIM prune up the tree toward the root node to prune (or remove) the unnecessary traffic. As this PIM prune travels hop by hop up the tree, each switch updates its forwarding state appropriately. Ultimately, the forwarding state associated with a multicast group or source is removed.

A multicast data sender sends data destined for a multicast group. The designated switch (DR) of the sender takes those data packets, unicast-encapsulates them, and sends them directly to the RP. The RP receives these encapsulated data packets, de-encapsulates them, and forwards them onto the shared tree. The packets then

follow the (*, G) multicast tree state in the switches on the RP tree, being replicated wherever the RP tree branches, and eventually reaching all the receivers for that multicast group. The process of encapsulating data packets to the RP is called registering, and the encapsulation packets are called PIM register packets.

IPv6 BSR: Configure RP Mapping

PIM switches in a domain must be able to map each multicast group to the correct RP address. The BSR protocol for PIM-SM provides a dynamic, adaptive mechanism to distribute group-to-RP mapping information rapidly throughout a domain. With the IPv6 BSR feature, if an RP becomes unreachable, it will be detected and the mapping tables will be modified so that the unreachable RP is no longer used, and the new tables will be rapidly distributed throughout the domain.

Every PIM-SM multicast group needs to be associated with the IP or IPv6 address of an RP. When a new multicast sender starts sending, its local DR will encapsulate these data packets in a PIM register message and send them to the RP for that multicast group. When a new multicast receiver joins, its local DR will send a PIM join message to the RP for that multicast group. When any PIM switch sends a (*, G) join message, the PIM switch needs to know which is the next switch toward the RP so that G (Group) can send a message to that switch. Also, when a PIM switch is forwarding data packets using (*, G) state, the PIM switch needs to know which is the correct incoming interface for packets destined for G, because it needs to reject any packets that arrive on other interfaces.

A small set of switches from a domain are configured as candidate bootstrap switches (C-BSRs) and a single BSR is selected for that domain. A set of switches within a domain are also configured as candidate RPs (C-RPs); typically, these switches are the same switches that are configured as C-BSRs. Candidate RPs periodically unicast candidate-RP-advertisement (C-RP-Adv) messages to the BSR of that domain, advertising their willingness to be an RP. A C-RP-Adv message includes the address of the advertising C-RP, and an optional list of group addresses and mask length fields, indicating the group prefixes for which the candidacy is advertised. The BSR then includes a set of these C-RPs, along with their corresponding group prefixes, in bootstrap messages (BSMs) it periodically originates. BSMs are distributed hop-by-hop throughout the domain.

Bidirectional BSR support allows bidirectional RPs to be advertised in C-RP messages and bidirectional ranges in the BSM. All switches in a system must be able to use the bidirectional range in the BSM; otherwise, the bidirectional RP feature will not function.

PIM-Source Specific Multicast

PIM-SSM is the routing protocol that supports the implementation of SSM and is derived from PIM-SM. However, unlike PIM-SM where data from all multicast sources are sent when there is a PIM join, the SSM feature forwards datagram traffic to receivers from only those multicast sources that the receivers have explicitly joined, thus optimizing bandwidth utilization and denying unwanted Internet broadcast traffic. Further, instead of the use of RP and shared trees, SSM uses information found on source addresses for a multicast group. This information is provided by receivers through the source addresses relayed to the last-hop switches by MLD membership reports, resulting in shortest-path trees directly to the sources.

In SSM, delivery of datagrams is based on (S, G) channels. Traffic for one (S, G) channel consists of datagrams with an IPv6 unicast source address S and the multicast group address G as the IPv6 destination address. Systems will receive this traffic by becoming members of the (S, G) channel. Signaling is not required, but receivers must subscribe or unsubscribe to (S, G) channels to receive or not receive traffic from specific sources.

MLD version 2 is required for SSM to operate. MLD allows the host to provide source information. Before SSM can run with MLD, SSM must be supported in the Cisco IOS IPv6 switch, the host where the application is running, and the application itself.

Routable Address Hello Option

When an IPv6 interior gateway protocol is used to build the unicast routing table, the procedure to detect the upstream switch address assumes the address of a PIM neighbor is always same as the address of the next-hop switch, as long as they refer to the same switch. However, it may not be the case when a switch has multiple addresses on a link.

Two typical situations can lead to this situation for IPv6. The first situation can occur when the unicast routing table is not built by an IPv6 interior gateway protocol such as multicast BGP. The second situation occurs when the address of an RP shares a subnet prefix with downstream switches (note that the RP switch address has to be domain-wide and therefore cannot be a link-local address).

The routable address hello option allows the PIM protocol to avoid such situations by adding a PIM hello message option that includes all the addresses on the interface on which the PIM hello message is advertised. When a PIM switch finds an upstream switch for some address, the result of RPF calculation is compared with the addresses in this option, in addition to the PIM neighbor's address itself. Because this option includes all the possible addresses of a PIM switch on that link, it always includes the RPF calculation result if it refers to the PIM switch supporting this option.

Because of size restrictions on PIM messages and the requirement that a routable address hello option fits within a single PIM hello message, a limit of 16 addresses can be configured on the interface.

PIM IPv6 Stub Routing

The PIM stub routing feature reduces resource usage by moving routed traffic closer to the end user.

In a network using PIM stub routing, the only allowable route for IPv6 traffic to the user is through a switch that is configured with PIM stub routing. PIM passive interfaces are connected to Layer 2 access domains, such as VLANs, or to interfaces that are connected to other Layer 2 devices. Only directly connected multicast receivers and sources are allowed in the Layer 2 access domains. The PIM passive interfaces do not send or process any received PIM control packets.

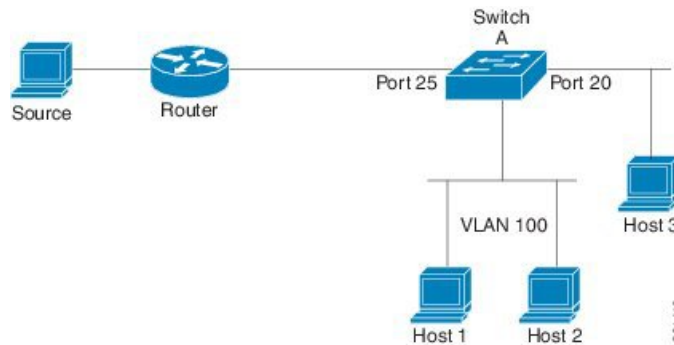
When using PIM stub routing, you should configure the distribution and remote routers to use IPv6 multicast routing and configure only the switch as a PIM stub router. The switch does not route transit traffic between distribution routers. You also need to configure a routed uplink port on the switch. The switch uplink port cannot be used with SVIs.

You must also configure EIGRP stub routing when configuring PIM stub routing on the switch.

The redundant PIM stub router topology is not supported. The redundant topology exists when there is more than one PIM router forwarding multicast traffic to a single access domain. PIM messages are blocked, and the PIM assert and designated router election mechanisms are not supported on the PIM passive interfaces. Only the non-redundant access router topology is supported by the PIM stub feature. By using a non-redundant topology, the PIM passive interface assumes that it is the only interface and designated router on that access domain.

In the figure shown below, Switch A routed uplink port 25 is connected to the router and PIM stub routing is enabled on the VLAN 100 interfaces and on Host 3. This configuration allows the directly connected hosts to receive traffic from multicast source.

Figure 17: PIM Stub Router Configuration



Rendezvous Point

IPv6 PIM provides embedded RP support. Embedded RP support allows the device to learn RP information using the multicast group destination address instead of the statically configured RP. For devices that are the RP, the device must be statically configured as the RP.

The device searches for embedded RP group addresses in MLD reports or PIM messages and data packets. On finding such an address, the device learns the RP for the group from the address itself. It then uses this learned RP for all protocol activity for the group. For devices that are the RP, the device is advertised as an embedded RP must be configured as the RP.

To select a static RP over an embedded RP, the specific embedded RP group range or mask must be configured in the access list of the static RP. When PIM is configured in sparse mode, you must also choose one or more devices to operate as an RP. An RP is a single common root placed at a chosen point of a shared distribution tree and is configured statically in each box.

PIM DRs forward data from directly connected multicast sources to the RP for distribution down the shared tree. Data is forwarded to the RP in one of two ways:

- Data is encapsulated in register packets and unicast directly to the RP by the first-hop device operating as the DR.
- If the RP has itself joined the source tree, it is multicast-forwarded per the RPF forwarding algorithm described in the PIM-Sparse Mode section.

The RP address is used by first-hop devices to send PIM register messages on behalf of a host sending a packet to the group. The RP address is also used by last-hop devices to send PIM join and prune messages to the RP to inform it about group membership. You must configure the RP address on all devices (including the RP device).

A PIM device can be an RP for more than one group. Only one RP address can be used at a time within a PIM domain for a certain group. The conditions specified by the access list determine for which groups the device is an RP.

IPv6 multicast supports the PIM accept register feature, which is the ability to perform PIM-SM register message filtering at the RP. The user can match an access list or compare the AS path for the registered source with the AS path specified in a route map.

Static Mroutes

IPv6 static mroutes behave much in the same way as IPv4 static mroutes used to influence the RPF check. IPv6 static mroutes share the same database as IPv6 static routes and are implemented by extending static route support for RPF checks. Static mroutes support equal-cost multipath mroutes, and they also support unicast-only static routes.

MRIB

The Multicast Routing Information Base (MRIB) is a protocol-independent repository of multicast routing entries instantiated by multicast routing protocols (routing clients). Its main function is to provide independence between routing protocols and the Multicast Forwarding Information Base (MFIB). It also acts as a coordination and communication point among its clients.

Routing clients use the services provided by the MRIB to instantiate routing entries and retrieve changes made to routing entries by other clients. Besides routing clients, MRIB also has forwarding clients (MFIB instances) and special clients such as MLD. MFIB retrieves its forwarding entries from MRIB and notifies the MRIB of any events related to packet reception. These notifications can either be explicitly requested by routing clients or spontaneously generated by the MFIB.

Another important function of the MRIB is to allow for the coordination of multiple routing clients in establishing multicast connectivity within the same multicast session. MRIB also allows for the coordination between MLD and routing protocols.

MFIB

The MFIB is a platform-independent and routing-protocol-independent library for IPv6 software. Its main purpose is to provide a Cisco IOS platform with an interface with which to read the IPv6 multicast forwarding table and notifications when the forwarding table changes. The information provided by the MFIB has clearly defined forwarding semantics and is designed to make it easy for the platform to translate to its specific hardware or software forwarding mechanisms.

When routing or topology changes occur in the network, the IPv6 routing table is updated, and those changes are reflected in the MFIB. The MFIB maintains next-hop address information based on the information in the IPv6 routing table. Because there is a one-to-one correlation between MFIB entries and routing table entries, the MFIB contains all known routes and eliminates the need for route cache maintenance that is associated with switching paths such as fast switching and optimum switching.

MFIB



Note Distributed MFIB has its significance only in a stacked environment where the active switch distributes the MFIB information to the other member switches in the stack. In the following section the line cards are nothing but the member switches in the stack.

MFIB (MFIB) is used to switch multicast IPv6 packets on distributed platforms. MFIB may also contain platform-specific information on replication across line cards. The basic MFIB routines that implement the core of the forwarding logic are common to all forwarding environments.

MFIB implements the following functions:

- Relays data-driven protocol events generated in the line cards to PIM.
- Provides an MFIB platform application program interface (API) to propagate MFIB changes to platform-specific code responsible for programming the hardware acceleration engine. This API also includes entry points to switch a packet in software (necessary if the packet is triggering a data-driven event) and to upload traffic statistics to the software.

The combination of MFIB and MRIB subsystems also allows the switch to have a "customized" copy of the MFIB database in each line card and to transport MFIB-related platform-specific information from the RP to the line cards.

IPv6 Multicast VRF Lite



Note The switch must be running the Network Advantage license to support IPv6 Multicast VRF lite.

The IPv6 Multicast VRF Lite feature provides IPv6 multicast support for multiple virtual routing/forwarding contexts (VRFs). The scope of these VRFs is limited to the device in which the VRFs are defined.

This feature provides separation between routing and forwarding, providing an additional level of security because no communication between devices belonging to different VRFs is allowed unless it is explicitly configured. The IPv6 Multicast VRF Lite feature simplifies the management and troubleshooting of traffic belonging to a specific VRF.

IPv6 Multicast Process Switching and Fast Switching

A unified MFIB is used to provide both fast switching and process switching support for PIM-SM and PIM-SSM in IPv6 multicast. In process switching, the switch must examine, rewrite, and forward each packet. The packet is first received and copied into the system memory. The switch then looks up the Layer 3 network address in the routing table. The Layer 2 frame is then rewritten with the next-hop destination address and sent to the outgoing interface. The switch also computes the cyclic redundancy check (CRC). This switching method is the least scalable method for switching IPv6 packets.

IPv6 multicast fast switching allows switches to provide better packet forwarding performance than process switching. Information conventionally stored in a route cache is stored in several data structures for IPv6 multicast switching. The data structures provide optimized lookup for efficient packet forwarding.

In IPv6 multicast forwarding, the first packet is fast-switched if the PIM protocol logic allows it. In IPv6 multicast fast switching, the MAC encapsulation header is precomputed. IPv6 multicast fast switching uses the MFIB to make IPv6 destination prefix-based switching decisions. In addition to the MFIB, IPv6 multicast fast switching uses adjacency tables to prepend Layer 2 addressing information. The adjacency table maintains Layer 2 next-hop addresses for all MFIB entries.

The adjacency table is populated as adjacencies are discovered. Each time an adjacency entry is created (such as through ARP), a link-layer header for that adjacent node is precomputed and stored in the adjacency table. Once a route is determined, it points to a next hop and corresponding adjacency entry. It is subsequently used for encapsulation during switching of packets.

A route might have several paths to a destination prefix, such as when a switch is configured for simultaneous load balancing and redundancy. For each resolved path, a pointer is added for the adjacency corresponding to the next-hop interface for that path. This mechanism is used for load balancing across several paths.

NSF and SSO Support In IPv6 Multicast

The switch running the Network Advantage license supports Non Stop Forwarding (NSF) and Stateful Switch Over (SSO) in IPv6.

For more information, see *Configuring NSF with SSO* topic in *Stack Manager and High Availability Configuration Guide* section.

NTP in IPv6

The Network Time Protocol (NTP) is a protocol designed to time-synchronize a network of machines. NTP runs over UDP, which in turn runs over IPv4. NTP Version 4 (NTPv4) is an extension of NTP version 3, which supports both IPv4 and IPv6.

For more information, see *Cisco IOS IPv6 Configuration Library* on Cisco.com.

How to Implement IPv6 Multicast

Enabling IPv6 Multicast Routing

To enable IPv6 multicast routing, perform this procedure:

Procedure

| | Command or Action | Purpose |
|---------------|---|--|
| Step 1 | enable Example: Device> enable | Enables privileged EXEC mode. Enter your password if prompted. |
| Step 2 | configure terminal Example: Device# configure terminal | Enter global configuration mode. |
| Step 3 | ipv6 multicast-routing Example: Device(config)# ipv6 multicast-routing | Enables multicast routing on all IPv6-enabled interfaces and enables multicast forwarding for PIM and MLD on all enabled interfaces of the switch. |
| Step 4 | copy running-config startup-config Example: Device(config)# copy running-config startup-config | (Optional) Save your entries in the configuration file. |

Customizing and Verifying the MLD Protocol

Customizing and Verifying MLD on an Interface

To customize and verify MLD on an interface, perform this procedure:

Procedure

| | Command or Action | Purpose |
|---------------|--|--|
| Step 1 | enable Example: Device> enable | Enables privileged EXEC mode. Enter your password if prompted. |
| Step 2 | configure terminal Example: Device# configure terminal | Enters global configuration mode. |
| Step 3 | interface <i>type number</i> Example: Device (config)# interface GigabitEthernet 1/0/1 | Specifies an interface type and number, and places the switch in interface configuration mode. |
| Step 4 | ipv6 mld join-group [<i>group-address</i>] [include exclude] { <i>source-address</i> source-list [<i>acl</i>] } Example: Device (config-if)# ipv6 mld join-group FF04::10 | Configures MLD reporting for a specified group and source. |
| Step 5 | ipv6 mld access-group <i>access-list-name</i> Example: Device (config-if)# ipv6 access-list acc-grp-1 | Allows the user to perform IPv6 multicast receiver access control. |
| Step 6 | ipv6 mld static-group [<i>group-address</i>] [include exclude] { <i>source-address</i> source-list [<i>acl</i>] } Example: Device (config-if)# ipv6 mld static-group ff04::10 include 100::1 | Statically forwards traffic for the multicast group onto a specified interface and cause the interface to behave as if a MLD joiner were present on the interface. |
| Step 7 | ipv6 mld query-max-response-time <i>seconds</i> Example: | Configures the timeout value before the switch takes over as the querier for the interface. |

| | Command or Action | Purpose |
|----------------|---|--|
| | Device(config-if)# ipv6 mld query-timeout 130 | |
| Step 8 | exit Example: Device(config-if)# exit | Enter this command twice to exit interface configuration mode and enter privileged EXEC mode. |
| Step 9 | show ipv6 mld groups [link-local] [group-name group-address] [interface-type interface-number] [detail explicit] Example: Device# show ipv6 mld groups GigabitEthernet 1/0/1 | Displays the multicast groups that are directly connected to the switch and that were learned through MLD. |
| Step 10 | show ipv6 mld groups summary Example: Device# show ipv6 mld groups summary | Displays the number of (*, G) and (S, G) membership reports present in the MLD cache. |
| Step 11 | show ipv6 mld interface [type number] Example: Device# show ipv6 mld interface GigabitEthernet 1/0/1 | Displays multicast-related information about an interface. |
| Step 12 | debug ipv6 mld [group-name group-address interface-type] Example: Device# debug ipv6 mld | Enables debugging on MLD protocol activity. |
| Step 13 | debug ipv6 mld explicit [group-name group-address] Example: Device# debug ipv6 mld explicit | Displays information related to the explicit tracking of hosts. |
| Step 14 | copy running-config startup-config | (Optional) Save your entries in the configuration file. |

Implementing MLD Group Limits

Per-interface and global MLD limits operate independently of each other. Both per-interface and global MLD limits can be configured on the same switch. The number of MLD limits, globally or per interface, is not configured by default; the limits must be configured by the user. A membership report that exceeds either the per-interface or the global state limit is ignored.

Implementing MLD Group Limits Globally

To implement MLD group limits globally, perform this procedure:

Procedure

| | Command or Action | Purpose |
|---------------|---|---|
| Step 1 | enable Example: Device> enable | Enables privileged EXEC mode. Enter your password if prompted. |
| Step 2 | configure terminal Example: Device# configure terminal | Enters global configuration mode. |
| Step 3 | ipv6 mld [vrf vrf-name] state-limit number Example: Device(config)# ipv6 mld state-limit 300 | Limits the number of MLD states globally. |
| Step 4 | copy running-config startup-config | (Optional) Save your entries in the configuration file. |

Implementing MLD Group Limits per Interface

To implement MLD group limits per interface, perform this procedure:

Procedure

| | Command or Action | Purpose |
|---------------|---|--|
| Step 1 | enable Example: Device> enable | Enables privileged EXEC mode. Enter your password if prompted. |
| Step 2 | configure terminal Example: Device# configure terminal | Enters global configuration mode. |
| Step 3 | interface type number Example: Device(config)# interface GigabitEthernet 1/0/1 | Specifies an interface type and number, and places the switch in interface configuration mode. |

| | Command or Action | Purpose |
|---------------|---|---|
| Step 4 | ipv6 mld limit <i>number</i> [except] <i>access-list</i> Example: Device(config-if)# ipv6 mld limit 100 | Limits the number of MLD states on a per-interface basis. |
| Step 5 | copy running-config startup-config | (Optional) Save your entries in the configuration file. |

Configuring Explicit Tracking of Receivers to Track Host Behavior

The explicit tracking feature allows a switch to track the behavior of the hosts within its IPv6 network and enables the fast leave mechanism to be used with MLD version 2 host reports.

To configuring explicit tracking of receivers to track host behavior, perform this procedure:

Procedure

| | Command or Action | Purpose |
|---------------|--|--|
| Step 1 | enable Example: Device> enable | Enables privileged EXEC mode. Enter your password if prompted. |
| Step 2 | configure terminal Example: Device# configure terminal | Enter global configuration mode. |
| Step 3 | interface <i>type number</i> Example: Device(config)# interface GigabitEthernet 1/0/1 | Specifies an interface type and number, and places the switch in interface configuration mode. |
| Step 4 | ipv6 mld explicit-tracking <i>access-list-name</i> Example: Device(config-if)# ipv6 mld explicit-tracking list1 | Enables explicit tracking of hosts. |
| Step 5 | copy running-config startup-config | (Optional) Save your entries in the configuration file. |

Resetting the MLD Traffic Counters

To reset the MLD traffic counters, perform this procedure:

Procedure

| | Command or Action | Purpose |
|---------------|---|---|
| Step 1 | enable Example: Device> enable | Enables privileged EXEC mode. Enter your password if prompted. |
| Step 2 | configure terminal Example: Device# configure terminal | Enters global configuration mode. |
| Step 3 | clear ipv6 mld traffic Example: Device# clear ipv6 mld traffic | Resets all MLD traffic counters. |
| Step 4 | show ipv6 mld traffic Example: Device# show ipv6 mld traffic | Displays the MLD traffic counters. |
| Step 5 | copy running-config startup-config | (Optional) Save your entries in the configuration file. |

Clearing the MLD Interface Counters

To clearing the MLD interface counters, perform this procedure

Procedure

| | Command or Action | Purpose |
|---------------|---|---|
| Step 1 | enable Example: Device> enable | Enables privileged EXEC mode. Enter your password if prompted. |
| Step 2 | configure terminal Example: Device# configure terminal | Enters global configuration mode. |
| Step 3 | clear ipv6 mld counters <i>interface-type</i> Example: | Clears the MLD interface counters. |

| | Command or Action | Purpose |
|---------------|--|---|
| | Device# <code>clear ipv6 mld counters Ethernet1/0</code> | |
| Step 4 | copy running-config startup-config | (Optional) Save your entries in the configuration file. |

Configuring PIM

This section explains how to configure PIM.

Configuring PIM-SM and Displaying PIM-SM Information for a Group Range

To configuring PIM-SM and view PIM-SM information for a group range, perform this procedure:

Procedure

| | Command or Action | Purpose |
|---------------|--|--|
| Step 1 | enable Example: Device> <code>enable</code> | Enables privileged EXEC mode. Enter your password if prompted. |
| Step 2 | configure terminal Example: Device# <code>configure terminal</code> | Enters global configuration mode. |
| Step 3 | ipv6 pim rp-address <i>ipv6-address[group-access-list]</i> Example: Device(config)# <code>ipv6 pim rp-address 2001:DB8::01:800:200E:8C6C acc-grp-1</code> | Configures the address of a PIM RP for a particular group range. |
| Step 4 | exit Example: Device(config)# <code>exit</code> | Exits global configuration mode, and returns the switch to privileged EXEC mode. |
| Step 5 | show ipv6 pim interface [state-on] [state-off] [type-number] Example: Device# <code>show ipv6 pim interface</code> | Displays information about interfaces configured for PIM. |

| | Command or Action | Purpose |
|----------------|--|---|
| Step 6 | show ipv6 pim group-map [<i>group-name</i> <i>group-address</i>] [<i>group-range</i> <i>group-mask</i>] [<i>info-source</i> { <i>bsr</i> default embedded-rp static }] Example: Device# show ipv6 pim group-map | Displays an IPv6 multicast group mapping table. |
| Step 7 | show ipv6 pim neighbor [detail] [<i>interface-type interface-number</i> count] Example: Device# show ipv6 pim neighbor | Displays the PIM neighbors discovered by the Cisco IOS software. |
| Step 8 | show ipv6 pim range-list [config] [<i>rp-address</i> <i>rp-name</i>] Example: Device# show ipv6 pim range-list | Displays information about IPv6 multicast range lists. |
| Step 9 | show ipv6 pim tunnel [<i>interface-type interface-number</i>] Example: Device# show ipv6 pim tunnel | Displays information about the PIM register encapsulation and de-encapsulation tunnels on an interface. |
| Step 10 | debug ipv6 pim [<i>group-name</i> <i>group-address</i> interface <i>interface-type</i> bsr group mvpn neighbor] Example: Device# debug ipv6 pim | Enables debugging on PIM protocol activity. |
| Step 11 | copy running-config startup-config | (Optional) Save your entries in the configuration file. |

Configuring PIM Options

To configure PIM options, perform this procedure:

Procedure

| | Command or Action | Purpose |
|---------------|---|---|
| Step 1 | enable Example: Device> enable | Enables privileged EXEC mode. Enter your password if prompted. |

| | Command or Action | Purpose |
|---------------|--|--|
| Step 2 | configure terminal Example: Device# configure terminal | Enters global configuration mode. |
| Step 3 | ipv6 pim spt-threshold infinity [group-list access-list-name] Example: Device(config)# ipv6 pim spt-threshold infinity group-list acc-grp-1 | Configures when a PIM leaf switch joins the SPT for the specified groups. |
| Step 4 | ipv6 pim accept-register {list access-list route-map map-name} Example: Device(config)# ipv6 pim accept-register route-map reg-filter | Accepts or rejects registers at the RP. |
| Step 5 | interface type number Example: Device(config)# interface GigabitEthernet 1/0/1 | Specifies an interface type and number, and places the switch in interface configuration mode. |
| Step 6 | ipv6 pim dr-priority value Example: Device(config-if)# ipv6 pim dr-priority 3 | Configures the DR priority on a PIM switch. |
| Step 7 | ipv6 pim hello-interval seconds Example: Device(config-if)# ipv6 pim hello-interval 45 | Configures the frequency of PIM hello messages on an interface. |
| Step 8 | ipv6 pim join-prune-interval seconds Example: Device(config-if)# ipv6 pim join-prune-interval 75 | Configures periodic join and prune announcement intervals for a specified interface. |
| Step 9 | exit Example: Device(config-if)# exit | Enter this command twice to exit interface configuration mode and enter privileged EXEC mode. |

| | Command or Action | Purpose |
|----------------|---|--|
| Step 10 | ipv6 pim join-prune statistic [<i>interface-type</i>] Example: Device(config-if)# show ipv6 pim join-prune statistic | Displays the average join-prune aggregation for the most recently aggregated packets for each interface. |
| Step 11 | copy running-config startup-config | (Optional) Save your entries in the configuration file. |

Resetting the PIM Traffic Counters

If PIM malfunctions or in order to verify that the expected number of PIM packets are received and sent, the user can clear PIM traffic counters. Once the traffic counters are cleared, the user can enter the `show ipv6 pim traffic` command to verify that PIM is functioning correctly and that PIM packets are being received and sent correctly.

To resetting the PIM traffic counters, perform this procedure:

Procedure

| | Command or Action | Purpose |
|---------------|---|---|
| Step 1 | enable Example: Device> enable | Enables privileged EXEC mode. Enter your password if prompted. |
| Step 2 | configure terminal Example: Device# configure terminal | Enters global configuration mode. |
| Step 3 | clear ipv6 pim traffic Example: Device# clear ipv6 pim traffic | Resets the PIM traffic counters. |
| Step 4 | show ipv6 pim traffic Example: Device# show ipv6 pim traffic | Displays the PIM traffic counters. |
| Step 5 | copy running-config startup-config | (Optional) Save your entries in the configuration file. |

Clearing the PIM Topology Table to Reset the MRIB Connection

No configuration is necessary to use the MRIB. However, users may in certain situations want to clear the PIM topology table in order to reset the MRIB connection and verify MRIB information.

To clear the PIM topology table to reset the MRIB connection, perform this procedure:

Procedure

| | Command or Action | Purpose |
|---------------|--|---|
| Step 1 | enable Example: Device> enable | Enables privileged EXEC mode. Enter your password if prompted. |
| Step 2 | configure terminal Example: Device# configure terminal | Enters global configuration mode. |
| Step 3 | clear ipv6 pim topology [<i>group-name</i> <i>group-address</i>] Example: Device# clear ipv6 pim topology FF04::10 | Clears the PIM topology table. |
| Step 4 | show ipv6 mrib client [<i>filter</i>] [<i>name {client-name client-name : client-id}</i>] Example: Device# show ipv6 mrib client | Displays multicast-related information about an interface. |
| Step 5 | show ipv6 mrib route { <i>link-local</i> <i>summary</i> } [<i>sourceaddress-or-name</i> *] [<i>groupname-or-address</i> [<i>prefix-length</i>]] Example: Device# show ipv6 mrib route | Displays the MRIB route information. |
| Step 6 | show ipv6 pim topology [<i>groupname-or-address</i> [<i>sourceaddress-or-name</i>] <i>link-local</i> <i>route-count</i> [<i>detail</i>]] Example: Device# show ipv6 pim topology | Displays PIM topology table information for a specific group or all groups. |

| | Command or Action | Purpose |
|----------------|--|---|
| Step 7 | debug ipv6 mrib client Example: Device# <code>debug ipv6 mrib client</code> | Enables debugging on MRIB client management activity. |
| Step 8 | debug ipv6 mrib io Example: Device# <code>debug ipv6 mrib io</code> | Enables debugging on MRIB I/O events. |
| Step 9 | debug ipv6 mrib proxy Example: Device# <code>debug ipv6 mrib proxy</code> | Enables debugging on MRIB proxy activity between the switch processor and line cards on distributed switch platforms. |
| Step 10 | debug ipv6 mrib route [<i>group-name</i> <i>group-address</i>] Example: Device# <code>debug ipv6 mrib route</code> | Displays information about MRIB routing entry-related activity. |
| Step 11 | debug ipv6 mrib table Example: Device# <code>debug ipv6 mrib table</code> | Enables debugging on MRIB table management activity. |
| Step 12 | copy running-config startup-config | (Optional) Save your entries in the configuration file. |

Configuring PIM IPv6 Stub Routing

The PIM Stub routing feature supports multicast routing between the distribution layer and the access layer. It supports two types of PIM interfaces, uplink PIM interfaces, and PIM passive interfaces. A routed interface configured with the PIM passive mode does not pass or forward PIM control traffic, it only passes and forwards MLD traffic.

PIM IPv6 Stub Routing Configuration Guidelines

- Before configuring PIM stub routing, you must have IPv6 multicast routing configured on both the stub router and the central router. You must also have PIM mode (sparse-mode) configured on the uplink interface of the stub router.
- The PIM stub router does not route the transit traffic between the distribution routers. Unicast (EIGRP) stub routing enforces this behavior. You must configure unicast stub routing to assist the PIM stub router behavior. For more information, see the *EIGRP Stub Routing* section.
- Only directly connected multicast (MLD) receivers and sources are allowed in the Layer 2 access domains. The PIM protocol is not supported in access domains.
- The redundant PIM stub router topology is not supported.

Default IPv6 PIM Routing Configuration

This table displays the default IPv6 PIM routing configuration for the .

Table 20: Default Multicast Routing Configuration

| Feature | Default Setting |
|-----------------------------------|-----------------------------|
| Multicast routing | Disabled on all interfaces. |
| PIM version | Version 2. |
| PIM mode | No mode is defined. |
| PIM stub routing | None configured. |
| PIM RP address | None configured. |
| PIM domain border | Disabled. |
| PIM multicast boundary | None. |
| Candidate BSRs | Disabled. |
| Candidate RPs | Disabled. |
| Shortest-path tree threshold rate | 0 kb/s. |
| PIM router query message interval | 30 seconds. |

Enabling IPV6 PIM Stub Routing

To enable IPV6 PIM stub routing, perform this procedure:

Before you begin

PIM stub routing is disabled in IPv6 by default.

Procedure

| | Command or Action | Purpose |
|---------------|---|---|
| Step 1 | enable Example: Device> enable | Enables privileged EXEC mode. Enter your password if prompted. |
| Step 2 | configure terminal Example: Device# configure terminal | Enters global configuration mode. |

| | Command or Action | Purpose |
|---------------|---|--|
| Step 3 | ipv6 multicast pim-passive-enable Example: <pre>Device(config-if) # ipv6 multicast pim-passive-enable</pre> | Enables IPv6 Multicast PIM routing on the switch. |
| Step 4 | interface interface-id Example: <pre>Device(config) # interface gigabitethernet 9/0/6</pre> | <p>Specifies the interface on which you want to enable PIM stub routing, and enters interface configuration mode.</p> <p>The specified interface must be one of the following:</p> <ul style="list-style-type: none"> • A routed port—A physical port that has been configured as a Layer 3 port by entering the no switchport interface configuration command. You will also need to enable IP PIM sparse mode on the interface, and join the interface as a statically connected member to an MLD static group. • An SVI—A VLAN interface created by using the interface vlan vlan-id global configuration command. You will also need to enable IP PIM sparse mode on the VLAN, join the VLAN as a statically connected member to an MLD static group, and then enable MLD snooping on the VLAN, the MLD static group, and physical interface. <p>These interfaces must have IPv6 addresses assigned to them.</p> |
| Step 5 | ipv6 pim Example: <pre>Device(config-if) # ipv6 pim</pre> | Enables the PIM on the interface. |
| Step 6 | ipv6 pim {bsr} {dr-priority value} {hello-interval seconds} {join-prune-interval seconds} {passive} Example: <pre>Device(config-if) # ipv6 pim bsr dr-priority hello-interval join-prune-interval passive</pre> | <p>Configures the various PIM stub features on the interface.</p> <p>Enter bsr to configure BSR on a PIM switch</p> <p>Enter dr-priority to configure the DR priority on a PIM switch.</p> <p>Enter hello-interval to configure the frequency of PIM hello messages on an interface.</p> |

| | Command or Action | Purpose |
|---------------|--|---|
| | | Enter join-prune-interval to configure periodic join and prune announcement intervals for a specified interface. Enter passive to configure the PIM in the passive mode. |
| Step 7 | end Example: Device(config-if)# end | Returns to privileged EXEC mode. |

Monitoring IPv6 PIM Stub Routing

Table 21: PIM Stub Configuration show Commands

| Command | Purpose |
|--------------------------------|--|
| show ipv6 pim interface | Displays the PIM stub that is enabled on each interface. |
| show ipv6 mld groups | Displays the interested clients that have joined the specific multicast source group. |
| show ipv6 mroute | Verifies that the multicast stream forwards from the source to the interested clients. |

Disabling Embedded RP Support in IPv6 PIM

A user might want to disable embedded RP support on an interface if all of the devices in the domain do not support embedded RP.



Note This task disables PIM completely, not just embedded RP support in IPv6 PIM.

To disabling embedded RP support in IPv6 PIM, perform this procedure:

Procedure

| | Command or Action | Purpose |
|---------------|---|---|
| Step 1 | enable Example: Device> enable | Enables privileged EXEC mode. Enter your password if prompted. |
| Step 2 | configure terminal Example: | Enters global configuration mode. |

| | Command or Action | Purpose |
|---------------|--|--|
| | Device# <code>configure terminal</code> | |
| Step 3 | no ipv6 pim [vrf vrf-name] rp embedded Example: Device(config)# <code>no ipv6 pim rp embedded</code> | Disables embedded RP support in IPv6 PIM. |
| Step 4 | interface type number Example: Device(config)# <code>interface FastEthernet 1/0</code> | Specifies an interface type and number, and places the device in interface configuration mode. |
| Step 5 | no ipv6 pim Example: Device(config-if)# <code>no ipv6 pim</code> | Turns off IPv6 PIM on a specified interface. |

Configuring a BSR

The tasks included here are described below.

Configuring a BSR and Verifying BSR Information

To configure and verify BSR Information, perform this procedure:

Procedure

| | Command or Action | Purpose |
|---------------|---|---|
| Step 1 | enable Example: Device> <code>enable</code> | Enables privileged EXEC mode. Enter your password if prompted. |
| Step 2 | configure terminal Example: Device# <code>configure terminal</code> | Enters global configuration mode. |
| Step 3 | ipv6 pim bsr candidate bsr <i>ipv6-address[hash-mask-length] [priority</i> <i>priority-value]</i> Example: | Configures a switch to be a candidate BSR. |

| | Command or Action | Purpose |
|---------------|--|--|
| | Device(config)# ipv6 pim bsr candidate bsr 2001:DB8:3000:3000::42 124 priority 10 | |
| Step 4 | interface <i>type number</i> Example: Device(config)# interface GigabitEthernet 1/0/1 | Specifies an interface type and number, and places the switch in interface configuration mode. |
| Step 5 | ipv6 pim bsr border Example: Device(config-if)# ipv6 pim bsr border | Specifies an interface type and number, and places the switch in interface configuration mode. |
| Step 6 | exit Example: Device(config-if)# exit | Enter this command twice to exit interface configuration mode and enter privileged EXEC mode. |
| Step 7 | show ipv6 pim bsr {election rp-cache candidate-rp} Example: Device(config-if)# show ipv6 pim bsr election | Displays information related to PIM BSR protocol processing. |
| Step 8 | copy running-config startup-config | (Optional) Save your entries in the configuration file. |

Sending PIM RP Advertisements to the BSR

To sending PIM RP advertisements to the BSR, perform this procedure:

Procedure

| | Command or Action | Purpose |
|---------------|---|---|
| Step 1 | enable Example: Device> enable | Enables privileged EXEC mode. Enter your password if prompted. |
| Step 2 | configure terminal Example: Device# configure terminal | Enters global configuration mode. |

| | Command or Action | Purpose |
|---------------|--|--|
| Step 3 | ipv6 pim bsr candidate rp <i>ipv6-address</i> [group-list <i>access-list-name</i>] [priority <i>priority-value</i>] [interval seconds] Example: Device(config)# ipv6 pim bsr candidate rp 2001:DB8:3000:3000::42 priority 0 | Sends PIM RP advertisements to the BSR. |
| Step 4 | interface <i>type number</i> Example: Device(config)# interface GigabitEthernet 1/0/1 | Specifies an interface type and number, and places the switch in interface configuration mode. |
| Step 5 | ipv6 pim bsr border Example: Device(config-if)# ipv6 pim bsr border | Configures a border for all BSMs of any scope on a specified interface. |
| Step 6 | copy running-config startup-config | (Optional) Save your entries in the configuration file. |

Configuring BSR for Use Within Scoped Zones

To configure BSR for use within scoped zones, perform this procedure:

Procedure

| | Command or Action | Purpose |
|---------------|--|---|
| Step 1 | enable Example: Device> enable | Enables privileged EXEC mode. Enter your password if prompted. |
| Step 2 | configure terminal Example: Device# configure terminal | Enters global configuration mode. |
| Step 3 | ipv6 pim bsr candidate rp <i>ipv6-address</i> [<i>hash-mask-length</i>] [priority <i>priority-value</i>] Example: Device(config)# ipv6 pim bsr candidate bsr 2001:DB8:1:1:4 | Configures a switch to be a candidate BSR. |

| | Command or Action | Purpose |
|---------------|--|--|
| Step 4 | ipv6 pim bsr candidate rp <i>ipv6-address</i> [group-list <i>access-list-name</i>] [priority <i>priority-value</i>] [interval seconds] Example: <pre>Device(config)# ipv6 pim bsr candidate rp 2001:DB8:1:1:1 group-list list scope 6</pre> | Configures the candidate RP to send PIM RP advertisements to the BSR. |
| Step 5 | interface <i>type number</i> Example: <pre>Device(config-if)# interface GigabitEthernet 1/0/1</pre> | Specifies an interface type and number, and places the switch in interface configuration mode. |
| Step 6 | ipv6 multicast boundary scope <i>scope-value</i> Example: <pre>Device(config-if)# ipv6 multicast boundary scope 6</pre> | Configures a multicast boundary on the interface for a specified scope. |
| Step 7 | copy running-config startup-config | (Optional) Save your entries in the configuration file. |

Configuring BSR Switches to Announce Scope-to-RP Mappings

IPv6 BSR switches can be statically configured to announce scope-to-RP mappings directly instead of learning them from candidate-RP messages. A user might want to configure a BSR switch to announce scope-to-RP mappings so that an RP that does not support BSR is imported into the BSR. Enabling this feature also allows an RP positioned outside the enterprise's BSR domain to be learned by the known remote RP on the local candidate BSR switch.

To configure BSR switches to announce Scope-to-RP mappings, perform this procedure:

Procedure

| | Command or Action | Purpose |
|---------------|---|---|
| Step 1 | enable Example: <pre>Device> enable</pre> | Enables privileged EXEC mode. Enter your password if prompted. |
| Step 2 | configure terminal Example: <pre>Device# configure terminal</pre> | Enters global configuration mode. |

| | Command or Action | Purpose |
|---------------|--|--|
| Step 3 | ipv6 pim bsr announced rp <i>ipv6-address</i> [group-list <i>access-list-name</i>] [priority <i>priority-value</i> Example: <pre>Device(config)# ipv6 pim bsr announced rp 2001:DB8:3000:3000::42 priority 0</pre> | Announces scope-to-RP mappings directly from the BSR for the specified candidate RP. |
| Step 4 | copy running-config startup-config | (Optional) Save your entries in the configuration file. |

Configuring SSM Mapping

When the SSM mapping feature is enabled, DNS-based SSM mapping is automatically enabled, which means that the switch will look up the source of a multicast MLD version 1 report from a DNS server.

You can use either DNS-based or static SSM mapping, depending on your switch configuration. If you choose to use static SSM mapping, you can configure multiple static SSM mappings. If multiple static SSM mappings are configured, the source addresses of all matching access lists will be used.



Note To use DNS-based SSM mapping, the switch needs to find at least one correctly configured DNS server, to which the switch may be directly attached.

To configuring SSM mapping, perform this procedure:

Procedure

| | Command or Action | Purpose |
|---------------|---|---|
| Step 1 | enable Example: <pre>Device> enable</pre> | Enables privileged EXEC mode. Enter your password if prompted. |
| Step 2 | configure terminal Example: <pre>Device# configure terminal</pre> | Enters global configuration mode. |
| Step 3 | ipv6 mld ssm-map enable Example: <pre>Device(config)# ipv6 mld ssm-map enable</pre> | Enables the SSM mapping feature for groups in the configured SSM range. |

| | Command or Action | Purpose |
|---------------|--|--|
| Step 4 | no ipv6 mld ssm-map query dns Example: Device(config)# no ipv6 mld ssm-map query dns | Disables DNS-based SSM mapping. |
| Step 5 | ipv6 mld ssm-map static <i>access-list source-address</i> Example: Device(config-if)# ipv6 mld ssm-map static SSM_MAP_ACL_2 2001:DB8:1::1 | Configures static SSM mappings. |
| Step 6 | exit Example: Device(config-if)# exit | Exits global configuration mode, and returns the switch to privileged EXEC mode. |
| Step 7 | show ipv6 mld ssm-map [<i>source-address</i>] Example: Device(config-if)# show ipv6 mld ssm-map | Displays SSM mapping information. |
| Step 8 | copy running-config startup-config | (Optional) Save your entries in the configuration file. |

Configuring Static Mroutes

Static multicast routes (mroutes) in IPv6 can be implemented as an extension of IPv6 static routes. You can configure your switch to use a static route for unicast routing only, to use a static multicast route for multicast RPF selection only, or to use a static route for both unicast routing and multicast RPF selection.

To configure static mroutes, perform this procedure:

Procedure

| | Command or Action | Purpose |
|---------------|---|---|
| Step 1 | enable Example: Device> enable | Enables privileged EXEC mode. Enter your password if prompted. |
| Step 2 | configure terminal Example: | Enters global configuration mode. |

| | Command or Action | Purpose |
|---------------|--|---|
| | Device# <code>configure terminal</code> | |
| Step 3 | <p><code>ipv6 route {ipv6-prefix / prefix-length ipv6-address interface-type interface-number ipv6-address} [administrative-distance] [administrative-multicast-distance unicast multicast] [tag tag]</code></p> <p>Example:</p> <pre>Device(config)# ipv6 route 2001:DB8::/64 6::6 100</pre> | Establishes static IPv6 routes. The example shows a static route used for both unicast routing and multicast RPF selection. |
| Step 4 | <p><code>exit</code></p> <p>Example:</p> <pre>Device# exit</pre> | Exits global configuration mode, and returns the switch to privileged EXEC mode. |
| Step 5 | <p><code>show ipv6 mroute [link-local [group-name group-address [source-address source-name]] [summary] [count]</code></p> <p>Example:</p> <pre>Device# show ipv6 mroute ff07::1</pre> | Displays the contents of the IPv6 multicast routing table. |
| Step 6 | <p><code>show ipv6 mroute [link-local group-name group-address] active [kpbs]</code></p> <p>Example:</p> <pre>Device(config-if)# show ipv6 mroute active</pre> | Displays the active multicast streams on the switch. |
| Step 7 | <p><code>show ipv6 rpf [ipv6-prefix]</code></p> <p>Example:</p> <pre>Device(config-if)# show ipv6 rpf 2001::1:1:2</pre> | Checks RPF information for a given unicast host address and prefix. |
| Step 8 | <code>copy running-config startup-config</code> | (Optional) Save your entries in the configuration file. |

Using MFIB in IPv6 Multicast

Multicast forwarding is automatically enabled when IPv6 multicast routing is enabled.

Verifying MFIB Operation in IPv6 Multicast

To verify MFIB operation in IPv6 multicast

Procedure

| | Command or Action | Purpose |
|---------------|--|---|
| Step 1 | enable Example: Device> enable | Enables privileged EXEC mode. Enter your password if prompted. |
| Step 2 | show ipv6 mfib [verbose <i>group-address-name</i> <i>ipv6-prefix/prefix-length</i> <i>source-address-name</i> count interface status summary] Example: Device# show ipv6 mfib | Displays the forwarding entries and interfaces in the IPv6 MFIB. |
| Step 3 | show ipv6 mfib [all linkscope group-name group-address [source-name source-address]] count Example: Device# show ipv6 mfib ff07::1 | Displays the contents of the IPv6 multicast routing table. |
| Step 4 | show ipv6 mfib interface Example: Device# show ipv6 mfib interface | Displays information about IPv6 multicast-enabled interfaces and their forwarding status. |
| Step 5 | show ipv6 mfib status Example: Device# show ipv6 mfib status | Displays general MFIB configuration and operational status. |
| Step 6 | show ipv6 mfib summary Example: Device# show ipv6 mfib summary | Displays summary information about the number of IPv6 MFIB entries and interfaces. |
| Step 7 | debug ipv6 mfib [<i>group-name</i> <i>group-address</i>] [adjacency db fs init interface mrrib [detail] nat pak platform ppr ps signal table] Example: Device# debug ipv6 mfib FF04::10 pak | Enables debugging output on the IPv6 MFIB. |

Resetting MFIB Traffic Counters

To reset MFIB traffic counters, perform this procedure:

Procedure

| | Command or Action | Purpose |
|---------------|--|---|
| Step 1 | enable Example: Device> enable | Enables privileged EXEC mode. Enter your password if prompted. |
| Step 2 | clear ipv6 mfib counters [<i>group-name</i> group-address [<i>source-address</i> <i>source-name</i>]] Example: Device# clear ipv6 mfib counters FF04::10 | Resets all active MFIB traffic counters. |

Additional References

Standards and RFCs

| Standard/RFC | Title |
|--------------------------|---|
| RFC 4292 | <i>IP Forwarding Table</i> |
| RFC 4293 | <i>Management Information Base for the Internet Protocol (IP)</i> |

Feature History for IPv6 Multicast

This table provides release and related information for the features explained in this module.

These features are available in all the releases subsequent to the one they were introduced in, unless noted otherwise.

| Release | Feature | Feature Information |
|-------------------------------|----------------|---|
| Cisco IOS XE Fuji 16.9.2 | IPv6 multicast | IPv6 multicast allows a host to send a single data stream to a subset of all hosts (group transmission) simultaneously |
| Cisco IOS XE Cupertino 17.9.1 | IPv6 multicast | This feature was implemented on C9200CX-12P-2X2G, C9200CX-8P-2X2G, and C9200CX-12T-2X2G models of the Cisco Catalyst 9200CX Series Switches, which were introduced in this release. |

Use the Cisco Feature Navigator to find information about platform and software image support. To access Cisco Feature Navigator, go to <http://www.cisco.com/go/cfn>.



CHAPTER 11

Configuring MLD Snooping

This module contains details of configuring MLD snooping

- [Information About Configuring IPv6 MLD Snooping, on page 227](#)
- [How to Configure IPv6 MLD Snooping, on page 230](#)
- [Monitoring MLD Snooping Information, on page 239](#)
- [Configuration Examples for Configuring MLD Snooping, on page 240](#)
- [Additional References, on page 241](#)
- [Feature History for MLD Snooping, on page 241](#)

Information About Configuring IPv6 MLD Snooping

You can use Multicast Listener Discovery (MLD) snooping to enable efficient distribution of IP Version 6 (IPv6) multicast data to clients and routers in a switched network on the switch. Unless otherwise noted, the term switch refers to a standalone switch and to a switch stack.

To use IPv6, you must configure the IPv6 Switch Database Management (SDM) template on the switch.

Understanding MLD Snooping

In IP Version 4 (IPv4), Layer 2 switches can use Internet Group Management Protocol (IGMP) snooping to limit the flooding of multicast traffic by dynamically configuring Layer 2 interfaces so that multicast traffic is forwarded to only those interfaces associated with IP multicast devices. In IPv6, MLD snooping performs a similar function. With MLD snooping, IPv6 multicast data is selectively forwarded to a list of ports that want to receive the data, instead of being flooded to all ports in a VLAN. This list is constructed by snooping IPv6 multicast control packets.

MLD is a protocol used by IPv6 multicast routers to discover the presence of multicast listeners (nodes wishing to receive IPv6 multicast packets) on the links that are directly attached to the routers and to discover which multicast packets are of interest to neighboring nodes. MLD is derived from IGMP; MLD Version 1 (MLDv1) is equivalent to IGMPv2, and MLD Version 2 (MLDv2) is equivalent to IGMPv3. MLD is a subprotocol of Internet Control Message Protocol Version 6 (ICMPv6), and MLD messages are a subset of ICMPv6 messages, identified in IPv6 packets by a preceding Next Header value of 58.

The switch supports two versions of MLD snooping:

- MLDv1 snooping detects MLDv1 control packets and sets up traffic bridging based on IPv6 destination multicast addresses.

- MLDv2 basic snooping (MBSS) uses MLDv2 control packets to set up traffic forwarding based on IPv6 destination multicast addresses.

The switch can snoop on both MLDv1 and MLDv2 protocol packets and bridge IPv6 multicast data based on destination IPv6 multicast addresses.



Note The switch does not support MLDv2 enhanced snooping, which sets up IPv6 source and destination multicast address-based forwarding.

MLD snooping can be enabled or disabled globally or per VLAN. When MLD snooping is enabled, a per-VLAN IPv6 multicast address table is constructed in software and hardware. The switch then performs IPv6 multicast-address based bridging in hardware.

MLD Messages

MLDv1 supports three types of messages:

- Listener Queries are the equivalent of IGMPv2 queries and are either General Queries or Multicast-Address-Specific Queries (MASQs).
- Multicast Listener Reports are the equivalent of IGMPv2 reports
- Multicast Listener Done messages are the equivalent of IGMPv2 leave messages.

MLDv2 supports MLDv2 queries and reports, as well as MLDv1 Report and Done messages.

Message timers and state transitions resulting from messages being sent or received are the same as those of IGMPv2 messages. MLD messages that do not have valid link-local IPv6 source addresses are ignored by MLD routers and switches.

MLD Queries

The switch sends out MLD queries, constructs an IPv6 multicast address database, and generates MLD group-specific and MLD group-and-source-specific queries in response to MLD Done messages. The switch also supports report suppression, report proxying, Immediate-Leave functionality, and static IPv6 multicast group address configuration.

When MLD snooping is disabled, all MLD queries are flooded in the ingress VLAN.

When MLD snooping is enabled, received MLD queries are flooded in the ingress VLAN, and a copy of the query is sent to the CPU for processing. From the received query, MLD snooping builds the IPv6 multicast address database. It detects multicast router ports, maintains timers, sets report response time, learns the querier IP source address for the VLAN, learns the querier port in the VLAN, and maintains multicast-address aging.

When a group exists in the MLD snooping database, the switch responds to a group-specific query by sending an MLDv1 report. When the group is unknown, the group-specific query is flooded to the ingress VLAN.

When a host wants to leave a multicast group, it can send out an MLD Done message (equivalent to IGMP Leave message). When the switch receives an MLDv1 Done message, if Immediate-Leave is not enabled, the switch sends an MASQ to the port from which the message was received to determine if other devices connected to the port should remain in the multicast group.

Multicast Client Aging Robustness

You can configure port membership removal from addresses based on the number of queries. A port is removed from membership to an address only when there are no reports to the address on the port for the configured number of queries. The default number is 2.

Multicast Router Discovery

Like IGMP snooping, MLD snooping performs multicast router discovery, with these characteristics:

- Ports configured by a user never age out.
- Dynamic port learning results from MLDv1 snooping queries and IPv6 PIMv2 packets.
- If there are multiple routers on the same Layer 2 interface, MLD snooping tracks a single multicast router on the port (the router that most recently sent a router control packet).
- Dynamic multicast router port aging is based on a default timer of 5 minutes; the multicast router is deleted from the router port list if no control packet is received on the port for 5 minutes.
- IPv6 multicast router discovery only takes place when MLD snooping is enabled on the switch.
- Received IPv6 multicast router control packets are always flooded to the ingress VLAN, whether or not MLD snooping is enabled on the switch.
- After the discovery of the first IPv6 multicast router port, unknown IPv6 multicast data is forwarded only to the discovered router ports (before that time, all IPv6 multicast data is flooded to the ingress VLAN).

MLD Reports

The processing of MLDv1 join messages is essentially the same as with IGMPv2. When no IPv6 multicast routers are detected in a VLAN, reports are not processed or forwarded from the switch. When IPv6 multicast routers are detected and an MLDv1 report is received, an IPv6 multicast group address is entered in the VLAN MLD database. Then all IPv6 multicast traffic to the group within the VLAN is forwarded using this address. When MLD snooping is disabled, reports are flooded in the ingress VLAN.

When MLD snooping is enabled, MLD report suppression, called listener message suppression, is automatically enabled. With report suppression, the switch forwards the first MLDv1 report received by a group to IPv6 multicast routers; subsequent reports for the group are not sent to the routers. When MLD snooping is disabled, report suppression is disabled, and all MLDv1 reports are flooded to the ingress VLAN.

The switch also supports MLDv1 proxy reporting. When an MLDv1 MASQ is received, the switch responds with MLDv1 reports for the address on which the query arrived if the group exists in the switch on another port and if the port on which the query arrived is not the last member port for the address.

MLD Done Messages and Immediate-Leave

When the Immediate-Leave feature is enabled and a host sends an MLDv1 Done message (equivalent to an IGMP leave message), the port on which the Done message was received is immediately deleted from the group. You enable Immediate-Leave on VLANs and (as with IGMP snooping), you should only use the feature on VLANs where a single host is connected to the port. If the port was the last member of a group, the group is also deleted, and the leave information is forwarded to the detected IPv6 multicast routers.

When Immediate Leave is not enabled in a VLAN (which would be the case when there are multiple clients for a group on the same port) and a Done message is received on a port, an MASQ is generated on that port.

The user can control when a port membership is removed for an existing address in terms of the number of MASQs. A port is removed from membership to an address when there are no MLDv1 reports to the address on the port for the configured number of queries.

The number of MASQs generated is configured by using the **ipv6 mld snooping last-listener-query count** global configuration command. The default number is 2.

The MASQ is sent to the IPv6 multicast address for which the Done message was sent. If there are no reports sent to the IPv6 multicast address specified in the MASQ during the switch maximum response time, the port on which the MASQ was sent is deleted from the IPv6 multicast address database. The maximum response time is the time configured by using the **ipv6 mld snooping last-listener-query-interval** global configuration command. If the deleted port is the last member of the multicast address, the multicast address is also deleted, and the switch sends the address leave information to all detected multicast routers.

Topology Change Notification Processing

When topology change notification (TCN) solicitation is enabled by using the **ipv6 mld snooping tcn query solicit** global configuration command, MLDv1 snooping sets the VLAN to flood all IPv6 multicast traffic with a configured number of MLDv1 queries before it begins sending multicast data only to selected ports. You set this value by using the **ipv6 mld snooping tcn flood query count** global configuration command. The default is to send two queries. The switch also generates MLDv1 global Done messages with valid link-local IPv6 source addresses when the switch becomes the STP root in the VLAN or when it is configured by the user. This is same as done in IGMP snooping.

How to Configure IPv6 MLD Snooping

Default MLD Snooping Configuration

Table 22: Default MLD Snooping Configuration

| Feature | Default Setting |
|----------------------------------|---|
| MLD snooping (Global) | Disabled. |
| MLD snooping (per VLAN) | Enabled. MLD snooping must be globally enabled for VLAN MLD snooping to take place. |
| IPv6 Multicast addresses | None configured. |
| IPv6 Multicast router ports | None configured. |
| MLD snooping Immediate Leave | Disabled. |
| MLD snooping robustness variable | Global: 2; Per VLAN: 0. Note The VLAN value overrides the global setting. When the VLAN value is 0, the VLAN uses the global count. |

| Feature | Default Setting |
|------------------------------|--|
| Last listener query count | Global: 2; Per VLAN: 0. Note The VLAN value overrides the global setting. When the VLAN value is 0, the VLAN uses the global count. |
| Last listener query interval | Global: 1000 (1 second); VLAN: 0. Note The VLAN value overrides the global setting. When the VLAN value is 0, the VLAN uses the global interval. |
| TCN query solicit | Disabled. |
| TCN query count | 2. |
| MLD listener suppression | Disabled. |

MLD Snooping Configuration Guidelines

When configuring MLD snooping, consider these guidelines:

- You can configure MLD snooping characteristics at any time, but you must globally enable MLD snooping by using the **ipv6 mld snooping** global configuration command for the configuration to take effect.
- MLD snooping and IGMP snooping act independently of each other. You can enable both features at the same time on the switch.
- The maximum number of multicast entries allowed for the switch or switch stack is 1024.

Enabling or Disabling MLD Snooping on the Switch

By default, IPv6 MLD snooping is globally disabled on the switch and enabled on all VLANs. When MLD snooping is globally disabled, it is also disabled on all VLANs. When you globally enable MLD snooping, the VLAN configuration overrides the global configuration. That is, MLD snooping is enabled only on VLAN interfaces in the default state (enabled).

You can enable and disable MLD snooping on a per-VLAN basis or for a range of VLANs, but if you globally disable MLD snooping, it is disabled in all VLANs. If global snooping is enabled, you can enable or disable VLAN snooping.

To globally enable MLD snooping on the switch, perform this procedure:

Procedure

| | Command or Action | Purpose |
|---------------|--------------------------------------|---|
| Step 1 | enable Example: | Enables privileged EXEC mode. Enter your password if prompted. |

| | Command or Action | Purpose |
|---------------|---|---|
| | Device> enable | |
| Step 2 | configure terminal Example: Device# configure terminal | Enters global configuration mode. |
| Step 3 | ipv6 mld snooping Example: Device(config)# ipv6 mld snooping | Enables MLD snooping on the switch. |
| Step 4 | end Example: Device(config)# end | Returns to privileged EXEC mode. |
| Step 5 | copy running-config startup-config Example: Device(config)# copy running-config startup-config | (Optional) Save your entries in the configuration file. |
| Step 6 | reload Example: Device(config)# reload | Reload the operating system. |

Enabling or Disabling MLD Snooping on a VLAN

To enable MLD snooping on a VLAN, perform this procedure:

Procedure

| | Command or Action | Purpose |
|---------------|---|---|
| Step 1 | enable Example: Device> enable | Enables privileged EXEC mode. Enter your password if prompted. |

| | Command or Action | Purpose |
|---------------|--|--|
| Step 2 | configure terminal Example: Device# <code>configure terminal</code> | Enters global configuration mode. |
| Step 3 | ipv6 mld snooping Example: Device(config)# <code>ipv6 mld snooping</code> | Enables MLD snooping on the switch. |
| Step 4 | ipv6 mld snooping vlan <i>vlan-id</i> Example: Device(config)# <code>ipv6 mld snooping vlan 1</code> | Enables MLD snooping on the VLAN. The VLAN ID range is 1 to 1001 and 1006 to 4094. Note MLD snooping must be globally enabled for VLAN snooping to be enabled. |
| Step 5 | end Example: Device(config)# <code>end</code> | Returns to privileged EXEC mode. |

Configuring a Static Multicast Group

Hosts or Layer 2 ports normally join multicast groups dynamically, but you can also statically configure an IPv6 multicast address and member ports for a VLAN.

To add a Layer 2 port as a member of a multicast group, perform this procedure:

Procedure

| | Command or Action | Purpose |
|---------------|---|--|
| Step 1 | enable Example: Device> <code>enable</code> | Enables privileged EXEC mode. Enter your password if prompted. |
| Step 2 | configure terminal Example: Device# <code>configure terminal</code> | Enters global configuration mode. |
| Step 3 | ipv6 mld snooping vlan <i>vlan-id</i> static <i>ipv6_multicast_address</i> interface <i>interface-id</i> | Configures a multicast group with a Layer 2 port as a member of a multicast group: |

| | Command or Action | Purpose |
|---------------|--|---|
| | Example: <pre>Device(config)# ipv6 mld snooping vlan 1 static 3333.0000.1111 interface gigabitethernet 1/ 1/0/1</pre> | <ul style="list-style-type: none"> • <i>vlan-id</i> is the multicast group VLAN ID. The VLAN ID range is 1 to 1001 and 1006 to 4094. • <i>ipv6_multicast_address</i> is the 128-bit group IPv6 address. The address must be in the form specified in RFC 2373. • <i>interface-id</i> is the member port. It can be a physical interface or a port channel (1 to 48). |
| Step 4 | end Example: <pre>Device(config)# end</pre> | Returns to privileged EXEC mode. |
| Step 5 | Use one of the following: <ul style="list-style-type: none"> • show ipv6 mld snooping address • show ipv6 mld snooping address vlan <i>vlan-id</i> Example: <pre>Device# show ipv6 mld snooping address OR Device# show ipv6 mld snooping vlan 1</pre> | Verifies the static member port and the IPv6 address. |

Configuring a Multicast Router Port



Note Static connections to multicast routers are supported only on switch ports.

To add a multicast router port to a VLAN, perform this procedure:

Procedure

| | Command or Action | Purpose |
|---------------|--|---|
| Step 1 | enable Example: <pre>Device> enable</pre> | Enables privileged EXEC mode. Enter your password if prompted. |

| | Command or Action | Purpose |
|---------------|--|--|
| Step 2 | configure terminal Example: Device# <code>configure terminal</code> | Enters global configuration mode. |
| Step 3 | ipv6 mld snooping vlan <i>vlan-id</i> mrouter interface <i>interface-id</i> Example: Device(config)# <code>ipv6 mld snooping vlan 1 mrouter interface gigabitethernet 1/0/2</code> | Specifies the multicast router VLAN ID, and specify the interface to the multicast router. <ul style="list-style-type: none"> • The VLAN ID range is 1 to 1001 and 1006 to 4094. • The interface can be a physical interface or a port channel. The port-channel range is 1 to 48. |
| Step 4 | end Example: Device(config)# <code>end</code> | Returns to privileged EXEC mode. |
| Step 5 | show ipv6 mld snooping mrouter [vlan <i>vlan-id</i>] Example: Device# <code>show ipv6 mld snooping mrouter vlan 1</code> | Verifies that IPv6 MLD snooping is enabled on the VLAN interface. |

Enabling MLD Immediate Leave

To enable MLDv1 immediate leave, perform this procedure:

Procedure

| | Command or Action | Purpose |
|---------------|---|---|
| Step 1 | enable Example: Device> <code>enable</code> | Enables privileged EXEC mode. Enter your password if prompted. |
| Step 2 | configure terminal Example: Device# <code>configure terminal</code> | Enters global configuration mode. |
| Step 3 | ipv6 mld snooping vlan <i>vlan-id</i> immediate-leave Example: | Enables MLD Immediate Leave on the VLAN interface. |

| | Command or Action | Purpose |
|---------------|--|---|
| | <code>Device(config)# ipv6 mld snooping vlan 1 immediate-leave</code> | |
| Step 4 | end Example: <code>Device(config)# end</code> | Returns to privileged EXEC mode. |
| Step 5 | show ipv6 mld snooping vlan <i>vlan-id</i> Example: <code>Device# show ipv6 mld snooping vlan 1</code> | Verifies that Immediate Leave is enabled on the VLAN interface. |

Configuring MLD Snooping Queries

To configure MLD snooping query characteristics for the switch or for a VLAN, perform this procedure:

Procedure

| | Command or Action | Purpose |
|---------------|---|---|
| Step 1 | enable Example: <code>Device> enable</code> | Enables privileged EXEC mode. Enter your password if prompted. |
| Step 2 | configure terminal Example: <code>Device# configure terminal</code> | Enters global configuration mode. |
| Step 3 | [no] ipv6 mld snooping robustness-variable <i>value</i> Example: <code>Device(config)# ipv6 mld snooping robustness-variable 3</code> | (Optional) Sets the number of queries that are sent before switch will deletes a listener (port) that does not respond to a general query. The range is 1 to 3; the default is 2. Use the no form of this command to disable this feature. |
| Step 4 | ipv6 mld snooping vlan <i>vlan-id</i> robustness-variable <i>value</i> Example: <code>Device(config)# ipv6 mld snooping vlan 1 robustness-variable 3</code> | (Optional) Sets the robustness variable on a VLAN basis, which determines the number of general queries that MLD snooping sends before aging out a multicast address when there is no MLD report response. The range is 1 to 3; the default is 0. When set to 0, the number that is used is the global robustness variable value. |

| | Command or Action | Purpose |
|---------|--|--|
| Step 5 | <p>[no] ipv6 mld snooping last-listener-query-count <i>count</i></p> <p>Example:</p> <pre>Device(config)# ipv6 mld snooping last-listener-query-count 7</pre> | <p>(Optional) Sets the number of MASQs that the switch sends before aging out an MLD client. The range is 1 to 7; the default is 2. The queries are sent 1 second apart.</p> <p>Use the no form of this command to disable this feature.</p> |
| Step 6 | <p>ipv6 mld snooping vlan <i>vlan-id</i> last-listener-query-count <i>count</i></p> <p>Example:</p> <pre>Device(config)# ipv6 mld snooping vlan 1 last-listener-query-count 7</pre> | <p>(Optional) Sets the last-listener query count on a VLAN basis. This value overrides the value that is configured globally. The range is 1 to 7; the default is 0. When set to 0, the global count value is used. Queries are sent 1 second apart.</p> |
| Step 7 | <p>[no] ipv6 mld snooping last-listener-query-interval <i>interval</i></p> <p>Example:</p> <pre>Device(config)# ipv6 mld snooping last-listener-query-interval 2000</pre> | <p>(Optional) Sets the maximum response time that the switch waits after sending out a MASQ before deleting a port from the multicast group. The range is 100 to 32,768 thousands of a second. The default is 1000 (1 second).</p> <p>Use the no form of this command to disable this feature.</p> |
| Step 8 | <p>ipv6 mld snooping vlan <i>vlan-id</i> last-listener-query-interval <i>interval</i></p> <p>Example:</p> <pre>Device(config)# ipv6 mld snooping vlan 1 last-listener-query-interval 2000</pre> | <p>(Optional) Sets the last-listener query interval on a VLAN basis. This value overrides the value that is configured globally. The range is 0 to 32,768 thousands of a second. The default is 0. When set to 0, the global last-listener query interval is used.</p> |
| Step 9 | <p>[no] ipv6 mld snooping tcn query solicit</p> <p>Example:</p> <pre>Device(config)# ipv6 mld snooping tcn query solicit</pre> | <p>(Optional) Enables topology change notification (TCN) solicitation, which means that VLANs flood all IPv6 multicast traffic for the configured number of queries before sending multicast data to only those ports requesting to receive it. The default is for TCN to be disabled.</p> <p>Use the no form of this command to disable this feature.</p> |
| Step 10 | <p>[no] ipv6 mld snooping tcn flood query count <i>count</i></p> <p>Example:</p> <pre>Device(config)# ipv6 mld snooping tcn flood query count 5</pre> | <p>(Optional) When TCN is enabled, specifies the number of TCN queries to be sent. The range is from 1 to 10; the default is 2.</p> <p>Use the no form of this command to disable this feature.</p> |
| Step 11 | <p>[no] ipv6 mld snooping querier</p> <p>Example:</p> | <p>(Optional) Enables MLD snooping queries.</p> |

| | Command or Action | Purpose |
|----------------|---|--|
| | Device (config) # ipv6 mld snooping querier | Note This command was introduced from Cisco IOS XE Cupertino 17.8.1 Use the no form of this command to disable MLD snooping queries. |
| Step 12 | [no] ipv6 mld snooping vlan <i>vlan_id</i> querier Example: Device (config) # ipv6 mld snooping vlan 1 querier | (Optional) Enables MLD snooping queries in a VLAN. Note This command was introduced from Cisco IOS XE Cupertino 17.8.1 Use the no form of this command to disable MLD snooping queries in a VLAN. |
| Step 13 | end Example: Device (config) # end | Returns to privileged EXEC mode. |
| Step 14 | show ipv6 mld snooping querier [vlan <i>vlan-id</i>] Example: Device# show ipv6 mld snooping querier vlan 1 | (Optional) Verifies that the MLD snooping querier information for the switch or for the VLAN. |

Disabling MLD Listener Message Suppression

MLD snooping listener message suppression is enabled by default. When it is enabled, the switch forwards only one MLD report per multicast router query. When message suppression is disabled, multiple MLD reports could be forwarded to the multicast routers.

To disable MLD listener message suppression, perform this procedure:

Procedure

| | Command or Action | Purpose |
|---------------|---|---|
| Step 1 | enable Example: Device> enable | Enables privileged EXEC mode. Enter your password if prompted. |
| Step 2 | configure terminal Example: | Enter global configuration mode. |

| | Command or Action | Purpose |
|---------------|---|---|
| | Device# <code>configure terminal</code> | |
| Step 3 | no ipv6 mld snooping listener-message-suppression Example: Device(config)# <code>no ipv6 mld snooping listener-message-suppression</code> | Disable MLD message suppression. |
| Step 4 | end Example: Device(config)# <code>end</code> | Return to privileged EXEC mode. |
| Step 5 | show ipv6 mld snooping Example: Device# <code>show ipv6 mld snooping</code> | Verify that IPv6 MLD snooping report suppression is disabled. |

Monitoring MLD Snooping Information

You can display MLD snooping information for dynamically learned and statically configured router ports and VLAN interfaces. You can also display IPv6 group address multicast entries for a VLAN configured for MLD snooping.

Table 23: Commands for Displaying MLD Snooping Information

| Command | Purpose |
|--|---|
| show ipv6 mld snooping [vlan <i>vlan-id</i>] | Displays the MLD snooping configuration information for all VLANs on the switch or for a specified VLAN. (Optional) Enter vlan <i>vlan-id</i> to display information for a single VLAN. The VLAN ID range is 1 to 1001 and 1006 to 4094. |
| show ipv6 mld snooping mrouter [vlan <i>vlan-id</i>] | Displays information on dynamically learned and manually configured multicast router interfaces. When you enable MLD snooping, the switch automatically learns the interface to which a multicast router is connected. These are dynamically learned interfaces. (Optional) Enters vlan <i>vlan-id</i> to display information for a single VLAN. The VLAN ID range is 1 to 1001 and 1006 to 4094. |
| show ipv6 mld snooping querier [vlan <i>vlan-id</i>] | Displays information about the IPv6 address and incoming port for the most-recently received MLD query messages in the VLAN. (Optional) Enters vlan <i>vlan-id</i> to display information for a single VLAN. The VLAN ID range is 1 to 1001 and 1006 to 4094. |

| Command | Purpose |
|--|---|
| show ipv6 mld snooping address [vlan <i>vlan-id</i>] [count dynamic user] | Displays all IPv6 multicast address information or specific IPv6 multicast address information for the switch or a VLAN. <ul style="list-style-type: none"> • Enters count to show the group count on the switch or in a VLAN. • Enters dynamic to display MLD snooping learned group information for the switch or for a VLAN. • Enters user to display MLD snooping user-configured group information for the switch or for a VLAN. |
| show ipv6 mld snooping address vlan <i>vlan-id</i> [<i>ipv6-multicast-address</i>] | Displays MLD snooping for the specified VLAN and IPv6 multicast address. |

Configuration Examples for Configuring MLD Snooping

Configuring a Static Multicast Group: Example

This example shows how to statically configure an IPv6 multicast group:

```
Device# configure terminal
Device(config)# ipv6 mld snooping vlan 2 static 3333.0000.1111 interface gigabitethernet1/0/1
Device(config)# end
```

Configuring a Multicast Router Port: Example

This example shows how to add a multicast router port to VLAN 200:

```
Device# configure terminal
Device(config)# ipv6 mld snooping vlan 200 mrouter interface gigabitethernet
1/0/2
Device(config)# exit
```

Enabling MLD Immediate Leave: Example

This example shows how to enable MLD Immediate Leave on VLAN 130:

```
Device# configure terminal
Device(config)# ipv6 mld snooping vlan 130 immediate-leave
Device(config)# exit
```

Configuring MLD Snooping Queries: Example

This example shows how to set the MLD snooping global robustness variable to 3:

```
Device# configure terminal
Device(config)# ipv6 mld snooping robustness-variable 3
Device(config)# exit
```

This example shows how to set the MLD snooping last-listener query count for a VLAN to 3:

```
Device# configure terminal
Device(config)# ipv6 mld snooping vlan 200 last-listener-query-count 3
Device(config)# exit
```

This example shows how to set the MLD snooping last-listener query interval (maximum response time) to 2000 (2 seconds):

```
Device# configure terminal
Device(config)# ipv6 mld snooping last-listener-query-interval 2000
Device(config)# exit
```

Additional References

Related Documents

| Related Topic | Document Title |
|--|--|
| For complete syntax and usage information for the commands used in this chapter. | <i>Command Reference (Catalyst 9200 Series Switches)</i> |

Feature History for MLD Snooping

This table provides release and related information for the features explained in this module.

These features are available in all the releases subsequent to the one they were introduced in, unless noted otherwise.

| Releases | Feature Name | Feature Information |
|-------------------------------|----------------------|---|
| Cisco IOS XE Fuji 16.9.2 | MLD Snooping | MLD snooping allows the switch to examine MLD packets and make forwarding decisions based on their content. |
| Cisco IOS XE Cupertino 17.8.1 | MLD Snooping Querier | The MLD snooping querier command enables MLD snooping queries in a VLAN. |

| Releases | Feature Name | Feature Information |
|-------------------------------|--------------------------------------|---|
| Cisco IOS XE Cupertino 17.9.1 | MLD Snooping MLD Snooping Querier | This feature was implemented on C9200CX-12P-2X2G, C9200CX-8P-2X2G, and C9200CX-12T-2X2G models of the Cisco Catalyst 9200CX Series Switches, which were introduced in this release. |

Use the Cisco Feature Navigator to find information about platform and software image support. To access Cisco Feature Navigator, go to <https://cfng.cisco.com/>



CHAPTER 12

IP Multicast Optimization: Optimizing PIM Sparse Mode in a Large IP Multicast Deployment

- [Prerequisites for Optimizing PIM Sparse Mode in a Large IP Multicast Deployment, on page 243](#)
- [Information About Optimizing PIM Sparse Mode in a Large IP Multicast Deployment, on page 243](#)
- [How to Optimize PIM Sparse Mode in a Large IP Multicast Deployment, on page 246](#)
- [Configuration Examples for Optimizing PIM Sparse Mode in a Large Multicast Deployment, on page 248](#)
- [Additional References for IP Multicast Optimization: Optimizing PIM Sparse Mode in a Large IP Multicast Deployment, on page 249](#)
- [Feature History for IP Multicast Optimization: Optimizing PIM Sparse Mode in a Large IP Multicast Deployment, on page 249](#)

Prerequisites for Optimizing PIM Sparse Mode in a Large IP Multicast Deployment

- You must have PIM sparse mode running in your network.
- If you plan to use a group list to control to which groups the shortest-path tree (SPT) threshold applies, you must have configured your access list before performing the task.

Information About Optimizing PIM Sparse Mode in a Large IP Multicast Deployment

PIM Registering Process

IP multicast sources do not use a signaling mechanism to announce their presence. Sources just send their data into the attached network, as opposed to receivers that use Internet Group Management Protocol (IGMP) to announce their presence. If a source sends traffic to a multicast group configured in PIM sparse mode (PIM-SM), the Designated Router (DR) leading toward the source must inform the rendezvous point (RP) about the presence of this source. If the RP has downstream receivers that want to receive the multicast traffic

(natively) from this source and has not joined the shortest path leading toward the source, then the DR must send the traffic from the source to the RP. The PIM registering process, which is individually run for each (S, G) entry, accomplishes these tasks between the DR and RP.

The registering process begins when a DR creates a new (S, G) state. The DR encapsulates all the data packets that match the (S, G) state into PIM register messages and unicasts those register messages to the RP.

If an RP has downstream receivers that want to receive register messages from a new source, the RP can either continue to receive the register messages through the DR or join the shortest path leading toward the source. By default, the RP will join the shortest path, because delivery of native multicast traffic provides the highest throughput. Upon receipt of the first packet that arrives natively through the shortest path, the RP will send a register-stop message back to the DR. When the DR receives this register-stop message, it will stop sending register messages to the RP.

If an RP has no downstream receivers that want to receive register messages from a new source, the RP will not join the shortest path. Instead, the RP will immediately send a register-stop message back to the DR. When the DR receives this register-stop message, it will stop sending register messages to the RP.

Once a routing entry is established for a source, a periodic reregistering takes place between the DR and RP. One minute before the multicast routing table state times out, the DR will send one dataless register message to the RP each second that the source is active until the DR receives a register-stop message from the RP.

This action restarts the timeout time of the multicast routing table entry, typically resulting in one reregistering exchange every 2 minutes. Reregistering is necessary to maintain state, to recover from lost state, and to keep track of sources on the RP. It will take place independently of the RP joining the shortest path.

PIM Version 1 Compatibility

If an RP is running PIM Version 1, it will not understand dataless register messages. In this case, the DR will not send dataless register messages to the RP. Instead, approximately every 3 minutes after receipt of a register-stop message from the RP, the DR encapsulates the incoming data packets from the source into register messages and sends them to the RP. The DR continues to send register messages until it receives another register-stop message from the RP. The same behavior occurs if the DR is running PIM Version 1.

When a DR running PIM Version 1 encapsulates data packets into register messages for a specific (S, G) entry, the entry is process-switched, not fast-switched or hardware-switched. On platforms that support these faster paths, the PIM registering process for an RP or DR running PIM Version 1 may lead to periodic out-of-order packet delivery. For this reason, we recommend upgrading your network from PIM Version 1 to PIM Version 2.

PIM Designated Router

Devices configured for IP multicast send PIM hello messages to determine which device will be the designated router (DR) for each LAN segment (subnet). The hello messages contain the device's IP address, and the device with the highest IP address becomes the DR.

The DR sends Internet Group Management Protocol (IGMP) host query messages to all hosts on the directly connected LAN. When operating in sparse mode, the DR sends source registration messages to the rendezvous point (RP).

By default, multicast devices send PIM router query messages every 30 seconds. By enabling a device to send PIM hello messages more often, the device can discover unresponsive neighbors more quickly. As a result, the device can implement failover or recovery procedures more efficiently. It is appropriate to make this change only on redundant devices on the edge of the network.

PIM Sparse-Mode Register Messages

Dataless register messages are sent at a rate of one message per second. Continuous high rates of register messages might occur if a DR is registering bursty sources (sources with high data rates) and if the RP is not running PIM Version 2.

By default, PIM sparse-mode register messages are sent without limiting their rate. Limiting the rate of register messages will limit the load on the DR and RP, at the expense of dropping those register messages that exceed the set limit. Receivers may experience data packet loss within the first second in which packets are sent from bursty sources.

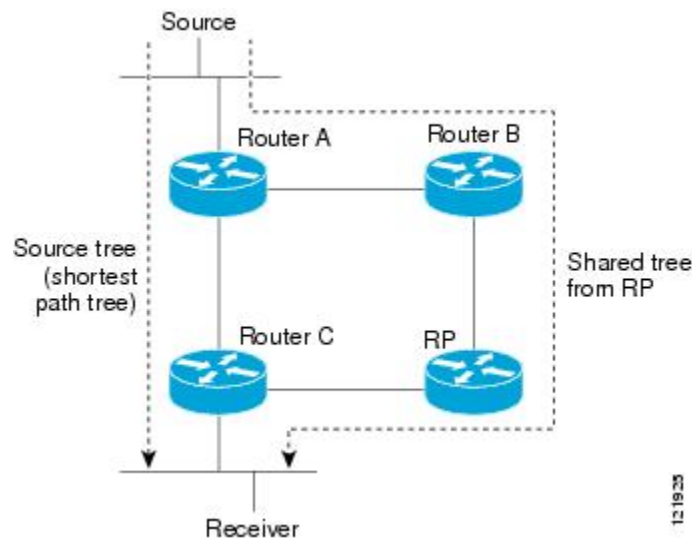
Preventing Use of Shortest-Path Tree to Reduce Memory Requirement

Understanding PIM shared tree and source tree will help you understand how preventing the use of the shortest-path tree can reduce memory requirements.

PIM Shared Tree and Source Tree - Shortest-Path Tree

By default, members of a multicast group receive data from senders to the group across a single data distribution tree rooted at the rendezvous point (RP). This type of distribution tree is called shared tree, as shown in the figure. Data from senders is delivered to the RP for distribution to group members joined to the shared tree.

Figure 18: Shared Tree versus Source Tree (Shortest-Path Tree)



If the data rate warrants, leaf routers on the shared tree may initiate a switch to the data distribution tree rooted at the source. This type of distribution tree is called a shortest-path tree (SPT) or source tree. By default, the software switches to a source tree upon receiving the first data packet from a source.

The following process describes the move from shared tree to source tree in more detail:

1. Receiver joins a group; leaf Router C sends a Join message toward the RP.
2. The RP puts the link to Router C in its outgoing interface list.
3. Source sends data; Router A encapsulates data in a register message and sends it to the RP.

4. The RP forwards data down the shared tree to Router C and sends a Join message toward the source. At this point, data may arrive twice at Router C, once encapsulated and once natively.
5. When data arrives natively (through multicast) at the RP, the RP sends a register-stop message to Router A.
6. By default, reception of the first data packet prompts Router C to send a Join message toward the source.
7. When Router C receives data on (S, G), it sends a Prune message for the source up the shared tree.
8. The RP deletes the link to Router C from the outgoing interface of (S, G). The RP triggers a Prune message toward the source.

Join and Prune messages are sent for sources and RPs. They are sent hop-by-hop and are processed by each PIM router along the path to the source or RP. Register and register-stop messages are not sent hop-by-hop. They are sent by the designated router that is directly connected to a source and are received by the RP for the group.

Multiple sources sending to groups use the shared tree.

Benefit of Preventing or Delaying the Use of the Shortest-Path Tree

The switch from shared to source tree happens upon the arrival of the first data packet at the last hop device (Router C in [PIM Shared Tree and Source Tree - Shortest-Path Tree, on page 245](#)). This switch occurs because the **ip pim spt-threshold** command controls that timing, and its default setting is 0 kbps.

The shortest-path tree requires more memory than the shared tree, but reduces delay. You might want to prevent or delay its use to reduce memory requirements. Instead of allowing the leaf device to move to the shortest-path tree immediately, you can prevent use of the SPT or specify that the traffic must first reach a threshold.

You can configure when a PIM leaf device should join the shortest-path tree for a specified group. If a source sends at a rate greater than or equal to the specified *kbps* rate, the device triggers a PIM Join message toward the source to construct a source tree (shortest-path tree). If the **infinity** keyword is specified, all sources for the specified group use the shared tree, never switching to the source tree.

How to Optimize PIM Sparse Mode in a Large IP Multicast Deployment

Optimizing PIM Sparse Mode in a Large Deployment

Consider performing this task if your deployment of IP multicast is large.

Steps 3, 5, and 6 in this task are independent of each other and are therefore considered optional. Any one of these steps will help optimize PIM sparse mode. If you are going to perform Step 5 or 6, you must perform Step 4. Step 6 applies only to a designated router; changing the PIM query interval is only appropriate on redundant routers on the edge of the PIM domain.

Procedure

| | Command or Action | Purpose |
|---------------|--|--|
| Step 1 | enable Example: <pre>Router> enable</pre> | Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted. |
| Step 2 | configure terminal Example: <pre>Router# configure terminal</pre> | Enters global configuration mode. |
| Step 3 | ip pim register-rate-limit <i>rate</i> Example: <pre>Router(config)# ip pim register-rate-limit 10</pre> | (Optional) Sets a limit on the maximum number of PIM sparse mode register messages sent per second for each (S, G) routing entry. <ul style="list-style-type: none"> • Use this command to limit the number of register messages that the designated router (DR) will allow for each (S, G) entry. • By default, there is no maximum rate set. • Configuring this command will limit the load on the DR and RP at the expense of dropping those register messages that exceed the set limit. • Receivers may experience data packet loss within the first second in which register messages are sent from bursty sources. |
| Step 4 | ip pim spt-threshold {<i>kpbs</i> infinity}[group-list <i>access-list</i>] Example: <pre>Router(config)# ip pim spt-threshold infinity group-list 5</pre> | (Optional) Specifies the threshold that must be reached before moving to the shortest-path tree. <ul style="list-style-type: none"> • The default value is 0, which causes the router to join the SPT immediately upon the first data packet it receives. • Specifying the infinity keyword causes the router never to move to the shortest-path tree; it remains on the shared tree. This keyword applies to a multicast environment of “many-to-many” communication. • The group list is a standard access list that controls which groups the SPT threshold applies to. If a value of 0 is specified or the group list is not used, the threshold applies to all groups. |

| | Command or Action | Purpose |
|---------------|---|--|
| | | <ul style="list-style-type: none"> In the example, group-list 5 is already configured to permit the multicast groups 239.254.2.0 and 239.254.3.0: access-list 5 permit 239.254.2.0 0.0.0.255 access-list 5 permit 239.254.3.0 0.0.0.255 |
| Step 5 | interface <i>type number</i> Example: <pre>Router(config)# interface GigabitEthernet 1/0/1</pre> | Configures an interface. <ul style="list-style-type: none"> If you do not want to change the default values of the PIM SPT threshold or the PIM query interval, do not perform this step; you are done with this task. |
| Step 6 | ip pim query-interval <i>period</i> [msec] Example: <pre>Router(config-if)# ip pim query-interval 1</pre> | (Optional) Configures the frequency at which multicast routers send PIM router query messages. <ul style="list-style-type: none"> Perform this step only on redundant routers on the edge of a PIM domain. The default query interval is 30 seconds. The <i>period</i> argument is in seconds unless the msec keyword is specified. Set the query interval to a smaller number of seconds for faster convergence, but keep in mind the trade-off between faster convergence and higher CPU and bandwidth usage. |

Configuration Examples for Optimizing PIM Sparse Mode in a Large Multicast Deployment

Optimizing PIM Sparse Mode in a Large IP Multicast Deployment Example

The following example shows how to:

- Set the query interval to 1 second for faster convergence.
- Configure the router to never move to the SPT but to remain on the shared tree.
- Set a limit of 10 PIM sparse mode register messages sent per second for each (S, G) routing entry.

```
interface GigabitEthernet 1/0/1
 ip pim query-interval 1
 .
 .
```

```

.
!
ip pim spt-threshold infinity
ip pim register-rate-limit 10
!

```

Additional References for IP Multicast Optimization: Optimizing PIM Sparse Mode in a Large IP Multicast Deployment

Related Documents

| Related Topic | Document Title |
|--|---|
| For complete syntax and usage information for the commands used in this chapter. | See the IP Multicast Routing Commands section of the <i>Command Reference (Catalyst 9200 Series Switches)</i> |

Feature History for IP Multicast Optimization: Optimizing PIM Sparse Mode in a Large IP Multicast Deployment

This table provides release and related information for the features explained in this module.

These features are available in all the releases subsequent to the one they were introduced in, unless noted otherwise.

| Release | Feature | Feature Information |
|--------------------------|--|--|
| Cisco IOS XE Fuji 16.9.2 | IP Multicast Optimization: Optimizing PIM Sparse Mode in a Large IP Multicast Deployment | Protocol Independent Multicast (PIM) has two basic operating modes: sparse-mode and dense-mode, and is suitable for large networks with heterogeneous links and devices. |

| Release | Feature | Feature Information |
|-------------------------------|--|---|
| Cisco IOS XE Cupertino 17.9.1 | IP Multicast Optimization: Optimizing PIM Sparse Mode in a Large IP Multicast Deployment | This feature was implemented on C9200CX-12P-2X2G, C9200CX-8P-2X2G, and C9200CX-12T-2X2G models of the Cisco Catalyst 9200CX Series Switches, which were introduced in this release. |

Use the Cisco Feature Navigator to find information about platform and software image support. To access Cisco Feature Navigator, go to <https://cfng.cisco.com/>



CHAPTER 13

IP Multicast Optimization: Multicast Subsecond Convergence

- [Prerequisites for Multicast Subsecond Convergence, on page 251](#)
- [Restrictions for Multicast Subsecond Convergence, on page 251](#)
- [Information About Multicast Subsecond Convergence, on page 251](#)
- [How to Configure Multicast Subsecond Convergence, on page 253](#)
- [Configuration Examples for Multicast Subsecond Convergence, on page 254](#)
- [Additional References for IP Multicast Optimization: Multicast Subsecond Convergence, on page 255](#)
- [Feature History for IP Multicast Optimization Multicast Subsecond Convergence, on page 255](#)

Prerequisites for Multicast Subsecond Convergence

Service providers must have a multicast-enabled core in order to use the Cisco Multicast Subsecond Convergence feature.

Restrictions for Multicast Subsecond Convergence

Devices that use the subsecond designated router (DR) failover enhancement must be able to process hello interval information arriving in milliseconds. Devices that are congested or do not have enough CPU cycles to process the hello interval can assume that the Protocol Independent Multicast (PIM) neighbor is disconnected, although this may not be the case.

Information About Multicast Subsecond Convergence

Benefits of Multicast Subsecond Convergence

- The scalability components improve on the efficiency of handling increases (or decreases) in service users (receivers) and service load (sources or content).
- New algorithms and processes (such as aggregated join messages, which deliver up to 1000 individual messages in a single packet) reduce the time to reach convergence by a factor of 10.

- Multicast subsecond convergence improves service availability for large multicast networks.
- Multicast users such as financial services firms and brokerages receive better quality of service (QoS), because multicast functionality is restored in a fraction of the time previously required.

Multicast Subsecond Convergence Scalability Enhancements

The Multicast Subsecond Convergence feature provides scalability enhancements that improve on the efficiency of handling increases (or decreases) in service users (receivers) and service load (sources or content). Scalability enhancements in this release include the following:

- Improved Internet Group Management Protocol (IGMP) and PIM state maintenance through new timer management techniques
- Improved scaling of the Multicast Source Discovery Protocol (MSDP) Source-Active (SA) cache

The scalability enhancements provide the following benefits:

- Increased potential PIM multicast route (mroute), IGMP, and MSDP SA cache state capacity
- Decreased CPU usage

PIM Router Query Messages

Multicast subsecond convergence allows you to send PIM router query messages (PIM hellos) every few milliseconds. The PIM hello message is used to locate neighboring PIM devices. Before the introduction of this feature, the device could send the PIM hellos only every few seconds. By enabling a device to send PIM hello messages more often, this feature allows the device to discover unresponsive neighbors more quickly. As a result, the device can implement failover or recovery procedures more efficiently.

Reverse Path Forwarding

Unicast Reverse Path Forwarding (RPF) helps to mitigate problems caused by the introduction of malformed or forged IP source addresses into a network by discarding IP packets that lack a verifiable IP source address. Malformed or forged source addresses can indicate denial-of-service (DoS) attacks based on source IP address spoofing.

RPF uses access control lists (ACLs) in determining whether to drop or forward data packets that have malformed or forged IP source addresses. An option in the ACL commands allows system administrators to log information about dropped or forwarded packets. Logging information about forged packets can help in uncovering information about possible network attacks.

Per-interface statistics can help system administrators quickly discover the interface serving as the entry point for an attack on the network.

Topology Changes and Multicast Routing Recovery

The Multicast Subsecond Convergence feature set enhances both enterprise and service provider network backbones by providing almost instantaneous recovery of multicast paths after unicast routing recovery.

Because PIM relies on the unicast routing table to calculate its RPF when a change in the network topology occurs, unicast protocols first need to calculate options for the best paths for traffic, and then multicast can determine the best path.

Multicast subsecond convergence allows multicast protocol calculations to finish almost immediately after the unicast calculations are completed. As a result, multicast traffic forwarding is restored substantially faster after a topology change.

How to Configure Multicast Subsecond Convergence

Modifying the PIM Router Query Message Interval

Perform this task to modify the PIM router query message interval.

Procedure

| | Command or Action | Purpose |
|---------------|--|---|
| Step 1 | enable Example: Device> enable | Enables privileged EXEC mode. • Enter your password if prompted. |
| Step 2 | configure terminal Example: Device# configure terminal | Enters global configuration mode. |
| Step 3 | interface type slot / subslot / port Example: Device(config)# interface gigabitethernet 1/0/0 | Specifies the interface and enters interface configuration mode. |
| Step 4 | ip pim query-interval period [msec] Example: Device(config-if)# ip pim query-interval 45 | Configures the frequency at which multicast routers send PIM router query messages. |

Verifying Multicast Subsecond Convergence Configurations

Perform this task to display detailed information about and to verify information regarding the Multicast Subsecond Convergence feature.

Procedure**Step 1** **enable****Example:**

```
Device> enable
```

Enables privileged EXEC mode.

- Enter your password if prompted.

Step 2 **show ip pim interface** *type number*

Use this command to display information about interfaces configured for PIM.

The following is sample output from the **show ip pim interface** command:

Example:

```
Device# show ip pim interface GigabitEthernet 1/0/0
Address          Interface          Ver/   Nbr   Query  DR      DR
                  Mode      Count  Intvl Prior
172.16.1.4      GigabitEthernet1/0/0  v2/S   1     100 ms 1       172.16.1.4
```

Step 3 **show ip pim neighbor**

Use this command to display the PIM neighbors discovered by the Cisco IOS XE software.

The following is sample output from the **show ip pim neighbor** command:

Example:

```
Device# show ip pim neighbor
PIM Neighbor Table
Neighbor      Interface          Uptime/Expires   Ver   DR
Address
172.16.1.3    GigabitEthernet1/0/0  00:03:41/250 msec v2    1 / S
```

Configuration Examples for Multicast Subsecond Convergence

Modifying the PIM Router Query Message Interval Example

In the following example, the **ip pim query-interval** command has been set to 100 milliseconds. This command does not show up in **show running-config** command output unless the interval value has been configured to be the nondefault value.

```
!
interface gigabitethernet 1/0/1
 ip address 172.16.2.1 255.255.255.0
 ip pim query-interval 100 msec
 ip pim sparse-mode
```


Additional References for IP Multicast Optimization: Multicast Subsecond Convergence

Related Documents

| Related Topic | Document Title |
|--|---|
| For complete syntax and usage information for the commands used in this chapter. | See the IP Multicast Routing Commands section of the <i>Command Reference (Catalyst 9200 Series Switches)</i> |

Feature History for IP Multicast Optimization Multicast Subsecond Convergence

This table provides release and related information for the features explained in this module.

These features are available in all the releases subsequent to the one they were introduced in, unless noted otherwise.

| Release | Feature | Feature Information |
|-------------------------------|--|---|
| Cisco IOS XE Fuji 16.9.2 | IP Multicast Optimization: Multicast Subsecond Convergence | The Multicast Subsecond Convergence feature provides scalability enhancements that improve on the efficiency of handling increases (or decreases) in service users (receivers) and service load (sources or content). |
| Cisco IOS XE Cupertino 17.9.1 | IP Multicast Optimization: Multicast Subsecond Convergence | This feature was implemented on C9200CX-12P-2X2G, C9200CX-8P-2X2G, and C9200CX-12T-2X2G models of the Cisco Catalyst 9200CX Series Switches, which were introduced in this release. |

Use the Cisco Feature Navigator to find information about platform and software image support. To access Cisco Feature Navigator, go to <https://cfmg.cisco.com/>



CHAPTER 14

IP Multicast Optimization: IP Multicast Load Splitting across Equal-Cost Paths

- [Prerequisites for IP Multicast Load Splitting across Equal-Cost Paths, on page 257](#)
- [Information About IP Multicast Load Splitting across Equal-Cost Paths, on page 257](#)
- [How to Load Split IP Multicast Traffic over ECMP, on page 265](#)
- [Configuration Examples for Load Splitting IP Multicast Traffic over ECMP, on page 271](#)
- [Additional References for IP Multicast Optimization: IP Multicast Load Splitting across Equal-Cost Paths, on page 272](#)
- [Feature History for IP Multicast Optimization: IP Multicast Load Splitting across Equal-Cost Paths, on page 272](#)

Prerequisites for IP Multicast Load Splitting across Equal-Cost Paths

IP multicast is enabled on the device using the tasks described in the “Configuring Basic IP Multicast” module of the *IP Multicast Routing Configuration Guide*.

Information About IP Multicast Load Splitting across Equal-Cost Paths

Load Splitting Versus Load Balancing

Load splitting and load balancing are not the same. Load splitting provides a means to randomly distribute (*, G) and (S, G) traffic streams across multiple equal-cost reverse path forwarding (RPF) paths, which does not necessarily result in a balanced IP multicast traffic load on those equal-cost RPF paths. By randomly distributing (*, G) and (S, G) traffic streams, the methods used for load splitting IP multicast traffic attempt to distribute an equal amount of traffic flows on each of the available RPF paths not by counting the flows, but, rather, by making a pseudorandom decision. These methods are collectively referred to as equal-cost multipath (ECMP) multicast load splitting methods and result in better load-sharing in networks where there are many traffic streams that utilize approximately the same amount of bandwidth.

If there are just a few (S, G) or (*, G) states flowing across a set of equal-cost links, the chance that they are well balanced is quite low. To overcome this limitation, precalculated source addresses--for (S, G) states or rendezvous point (RP) addresses for (*, G) states, can be used to achieve a reasonable form of load balancing. This limitation applies equally to the per-flow load splitting in Cisco Express Forwarding (CEF) or with EtherChannels: As long as there are only a few flows, those methods of load splitting will not result in good load distribution without some form of manual engineering.

Default Behavior for IP Multicast When Multiple Equal-Cost Paths Exist

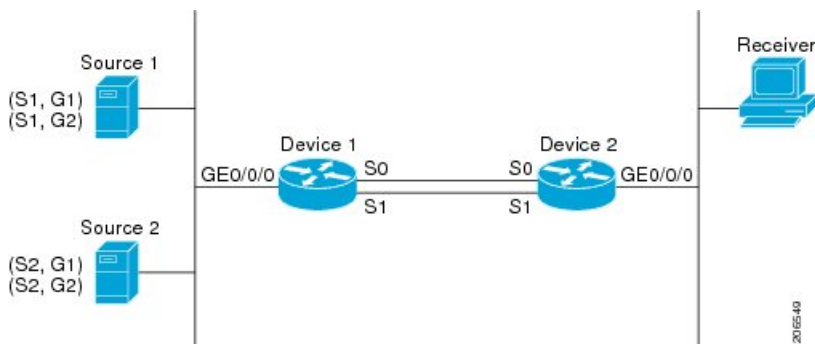
By default, for Protocol Independent Multicast sparse mode (PIM-SM), Source Specific Multicast (PIM-SSM), bidirectional PIM (bidir-PIM), groups, if multiple equal-cost paths are available, Reverse Path Forwarding (RPF) for IPv4 multicast traffic is based on the PIM neighbor with the highest IP address. This method is referred to as the highest PIM neighbor behavior. This behavior is in accordance with RFC 2362 for PIM-SM, but also applies to PIM-SSM, and bidir-PIM.

The figure illustrates a sample topology that is used in this section to explain the default behavior for IP multicast when multiple equal-cost paths exist.



Note Although the following illustration and example uses routers in the configuration, any device (router or controller) can be used.

Figure 19: Default Behavior for IP Multicast When Multiple Equal-Cost Paths Exist



In the figure, two sources, S1 and S2, are sending traffic to IPv4 multicast groups, G1 and G2. Either PIM-SM, PIM-SSM can be used in this topology. If PIM-SM is used, assume that the default of 0 for the **ip pim spt-threshold** command is being used on Device 2, that an Interior Gateway Protocol (IGP) is being run, and that the output of the **show ip route** command for S1 and for S2 (when entered on Device 2) displays serial interface 0 and serial interface 1 on Device 1 as equal-cost next-hop PIM neighbors of Device 2.

Without further configuration, IPv4 multicast traffic in the topology illustrated in the figure would always flow across one serial interface (either serial interface 0 or serial interface 1), depending on which interface has the higher IP address. For example, suppose that the IP addresses configured on serial interface 0 and serial interface 1 on Device 1 are 10.1.1.1 and 10.1.2.1, respectively. Given that scenario, in the case of PIM-SM and PIM-SSM, Device 2 would always send PIM join messages towards 10.1.2.1 and would always receive IPv4 multicast traffic on serial interface 1 for all sources and groups shown in the figure.

IPv4 RPF lookups are performed by intermediate multicast device to determine the RPF interface and RPF neighbor for IPv4 (*,G) and (S, G) multicast routes (trees). An RPF lookup consists of RPF route-selection and route-path-selection. RPF route-selection operates solely on the IP unicast address to identify the root of

the multicast tree. For (*, G) routes (PIM-SM and Bidir-PIM), the root of the multicast tree is the RP address for the group G; for (S, G) trees (PIM-SM, PIM-SSM), the root of the multicast tree is the source S. RPF route-selection finds the best route towards the RP or source in the routing information base (RIB), and, if configured (or available), the Distance Vector Multicast Routing Protocol (DVMRP) routing table or configured static mroutes. If the resulting route has only one available path, then the RPF lookup is complete, and the next-hop device and interface of the route become the RPF neighbor and RPF interface of this multicast tree. If the route has more than one path available, then route-path-selection is used to determine which path to choose.

For IP multicast, the following route-path-selection methods are available:



Note All methods but the default method of route-path-selection available in IP multicast enable some form of ECMP multicast load splitting.

- Highest PIM neighbor--This is the default method; thus, no configuration is required. If multiple equal-cost paths are available, RPF for IPv4 multicast traffic is based on the PIM neighbor with the highest IP address; as a result, without configuration, ECMP multicast load splitting is disabled by default.
- ECMP multicast load splitting method based on source address--You can configure ECMP multicast load splitting using the **ip multicast multipath** command. Entering this form of the **ip multicast multipath** command enables ECMP multicast load splitting based on source address using the S-hash algorithm. For more information, see the *ECMP Multicast Load Splitting Based on Source Address Using the S-Hash Algorithm* section.
- ECMP multicast load splitting method based on source and group address--You can configure ECMP multicast load splitting using the **ip multicast multipath** command with the **s-g-hash** and **basic** keywords. Entering this form of the **ip multicast multipath** command enables ECMP multicast load splitting based on source and group address using the basic S-G-hash algorithm. For more information, see the *ECMP Multicast Load Splitting Based on Source and Group Address Using the Basic S-G-Hash Algorithm* section.
- ECMP multicast load splitting method based on source, group, and next-hop address--You can configure ECMP multicast load splitting using the **ip multicast multipath** command with the **s-g-hash** and **next-hop-based** keywords. Entering this form of the command enables ECMP multicast load splitting based on source, group, and next-hop address using the next-hop-based S-G-hash algorithm. For more information, see the *ECMP Multicast Load Splitting Based on Source Group and Next-Hop Address* section.

The default behavior (the highest PIM neighbor behavior) does not result in any form of ECMP load-splitting in IP multicast, but instead selects the PIM neighbor that has the highest IP address among the next-hop PIM neighbors for the available paths. A next hop is considered to be a PIM neighbor when it displays in the output of the **show ip pim neighbor** command, which is the case when PIM hello messages have been received from it and have not timed out. If none of the available next hops are PIM neighbors, then simply the next hop with the highest IP address is chosen.

Methods to Load Split IP Multicast Traffic

In general, the following methods are available to load split IP multicast traffic:

- You can enable ECMP multicast load splitting based on source address, based on source and group address, or based on source, group, and next-hop address. After the equal-cost paths are recognized,

ECMP multicast load splitting operates on a per (S, G) basis, rather than a per packet basis as in unicast traffic.

- Alternative methods to load split IP multicast are to consolidate two or more equal-cost paths into a generic routing encapsulation (GRE) tunnel and allow the unicast routing protocol to perform the load splitting, or to load split across bundle interfaces, such as Fast or Gigabit EtherChannel interfaces, Multilink PPP (MLPPP) link bundles, or Multilink Frame Relay (FR.16) link bundles.

Overview of ECMP Multicast Load Splitting

By default, ECMP multicast load splitting of IPv4 multicast traffic is disabled. ECMP multicast load splitting can be enabled using the `ip multicast multipath` command.

ECMP Multicast Load Splitting Based on Source Address Using the S-Hash Algorithm

ECMP multicast load splitting traffic based on source address uses the S-hash algorithm, enabling the RPF interface for each (*, G) or (S, G) state to be selected among the available equal-cost paths, depending on the RPF address to which the state resolves. For an (S, G) state, the RPF address is the source address of the state; for a (*, G) state, the RPF address is the address of the RP associated with the group address of the state.

When ECMP multicast load splitting based on source address is configured, multicast traffic for different states can be received across more than just one of the equal-cost interfaces. The method applied by IPv4 multicast is quite similar in principle to the default per-flow load splitting in IPv4 CEF or the load splitting used with Fast and Gigabit EtherChannels. This method of ECMP multicast load splitting, however, is subject to polarization.

ECMP Multicast Load Splitting Based on Source and Group Address Using the Basic S-G-Hash Algorithm

ECMP multicast load splitting based on source and group address uses a simple hash, referred to as the basic S-G-hash algorithm, which is based on source and group address. The basic S-G-hash algorithm is predictable because no randomization is used in coming up with the hash value. The S-G-hash mechanism, however, is subject to polarization because for a given source and group, the same hash is always picked irrespective of the device this hash is being calculated on.



Note The basic S-G-hash algorithm ignores bidir-PIM groups.

Predictability As a By-Product of Using the S-Hash and Basic S-G-Hash Algorithms

The method used by ECMP multicast load splitting in IPv4 multicast allows for consistent load splitting in a network where the same number of equal-cost paths are present in multiple places in a topology. If an RP address or source addresses are calculated once to have flows split across N paths, then they will be split across those N paths in the same way in all places in the topology. Consistent load splitting allows for predictability, which, in turn, enables load splitting of IPv4 multicast traffic to be manually engineered.

Polarization As a By-Product of Using the S-Hash and Basic S-G-Hash Algorithms

The hash mechanism used in IPv4 multicast to load split multicast traffic by source address or by source and group address is subject to a problem usually referred to as polarization. A by-product of ECMP multicast

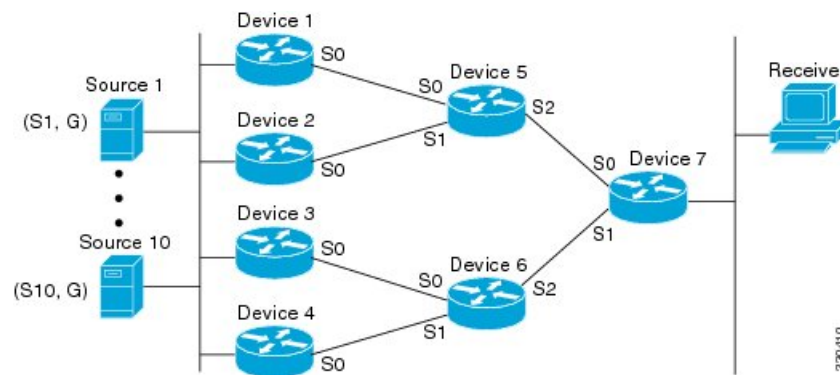
load splitting based on source address or on source and group address, polarization is a problem that prevents routers in some topologies from effectively utilizing all available paths for load splitting.

The figure illustrates a sample topology that is used in this section to explain the problem of polarization when configuring ECMP multicast load splitting based on source address or on source and group address.



Note Although the following illustration and example uses routers in the configuration, any device (router or switch) can be used.

Figure 20: Polarization Topology



In the topology illustrated in the figure, notice that Router 7 has two equal-cost paths towards the sources, S1 to S10, through Router 5 and Router 6. For this topology, suppose that ECMP multicast load splitting is enabled with the **ip multicast multipath** command on all routers in the topology. In that scenario, Router 7 would apply equal-cost load splitting to the 10 (S, G) states. The problem of polarization in this scenario would affect Router 7 because that router would end up choosing serial interface 0 on Router 5 for sources S1 to S5 and serial interface 1 on Router 6 for sources S6 to S10. The problem of polarization, furthermore, would also affect Router 5 and Router 6 in this topology. Router 5 has two equal-cost paths for S1 to S5 through serial interface 0 on Router 1 and serial interface 1 on Router 2. Because Router 5 would apply the same hash algorithm to select which of the two paths to use, it would end up using just one of these two upstream paths for sources S1 to S5; that is, either all the traffic would flow across Router 1 and Router 5 or across Router 2 and Router 5. It would be impossible in this topology to utilize Router 1 and Router 5 and Router 2 and Router 5 for load splitting. Likewise, the polarization problem would apply to Router 3 and Router 6 and Router 4 and Router 6; that is, it would be impossible in this topology to utilize both Router 3 and Router 6 and Router 4 and Router 6 for load splitting.

ECMP Multicast Load Splitting Based on Source Group and Next-Hop Address

Configuring ECMP multicast load splitting based on source, group, and next-hop address enables a more complex hash, the next-hop-based S-G-hash algorithm, which is based on source, group, and next-hop address. The next-hop-based S-G-hash algorithm is predictable because no randomization is used in calculating the hash value. Unlike the S-hash and basic S-G-hash algorithms, the hash mechanism used by the next-hop-based S-G-hash algorithm is not subject to polarization.



Note The next-hop-based S-G-hash algorithm in IPv4 multicast is the same algorithm used in IPv6 ECMP multicast load splitting, which, in turn, utilizes the same hash function used for PIM-SM bootstrap device (BSR).

The next-hop-based hash mechanism does not produce polarization and also maintains better RPF stability when paths fail. These benefits come at the cost that the source or RP IP addresses cannot be used to reliably predict and engineer the outcome of load splitting when the next-hop-based S-G-hash algorithm is used. Because many customer networks have implemented equal-cost multipath topologies, the manual engineering of load splitting, thus, is not a requirement in many cases. Rather, it is more of a requirement that the default behavior of IP multicast be similar to IP unicast; that is, it is expected that IP multicast use multiple equal-cost paths on a best-effort basis. Load splitting for IPv4 multicast, therefore, could not be enabled by default because of the anomaly of polarization.



Note Load splitting for CEF unicast also uses a method that does not exhibit polarization and likewise cannot be used to predict the results of load splitting or engineer the outcome of load splitting.

The next-hop-based hash function avoids polarization because it introduces the actual next-hop IP address of PIM neighbors into the calculation, so the hash results are different for each device, and in effect, there is no problem of polarization. In addition to avoiding polarization, this hash mechanism also increases stability of the RPF paths chosen in the face of path failures. Consider a device with four equal-cost paths and a large number of states that are load split across these paths. Suppose that one of these paths fails, leaving only three available paths. With the hash mechanism used by the polarizing hash mechanisms (the hash mechanism used by the S-hash and basic S-G-hash algorithms), the RPF paths of all states would likely reconverge and thus change between those three paths, especially those paths that were already using one of those three paths. These states, therefore, may unnecessarily change their RPF interface and next-hop neighbor. This problem exists simply because the chosen path is determined by taking the total number of paths available into consideration by the algorithm, so once a path changes, the RPF selection for all states is subject to change too. For the next-hop-based hash mechanism, only the states that were using the changed path for RPF would need to reconverge onto one of the three remaining paths. The states that were already using one of those paths would not change. If the fourth path came back up, the states that initially used it would immediately reconverge back to that path without affecting the other states.



Note The next-hop-based S-G-hash algorithm ignores bidir-PIM groups.

Effect of ECMP Multicast Load Splitting on PIM Neighbor Query and Hello Messages for RPF Path Selection

If load splitting of IP multicast traffic over ECMP is not enabled and there are multiple equal-cost paths towards an RP or a source, IPv4 multicast will first elect the highest IP address PIM neighbor. A PIM neighbor is a device from which PIM hello (or PIMv1 query) messages are received. For example, consider a device that has two equal-cost paths learned by an IGP or configured through two static routes. The next hops of these two paths are 10.1.1.1 and 10.1.2.1. If both of these next-hop devices send PIM hello messages, then 10.1.2.1 would be selected as the highest IP address PIM neighbor. If only 10.1.1.1 sends PIM hello messages, then 10.1.1.1 would be selected. If neither of these devices sends PIM hello messages, then 10.1.2.1 would be selected. This deference to PIM hello messages allows the construction of certain types of dynamic failover scenarios with only static multicast routes (mroutes); it is otherwise not very useful.



Note For more information about configuring static mroutes, see the *Configuring Multiple Static Mroutes in Cisco IOS* configuration note on the Cisco IOS IP multicast FTP site, which is available at: [ftp://ftpeng.cisco.com/ipmulticast/config-notes/static-mroutes.txt](http://ftpeng.cisco.com/ipmulticast/config-notes/static-mroutes.txt).

When load splitting of IP multicast traffic over ECMP is enabled, the presence of PIM hello message from neighbors is not considered; that is, the chosen RPF neighbor does not depend on whether or not PIM hello messages are received from that neighbor--it only depends on the presence or absence of an equal-cost route entry.

Effect of ECMP Multicast Load Splitting on the PIM Assert Process in PIM-SM and PIM-SSM

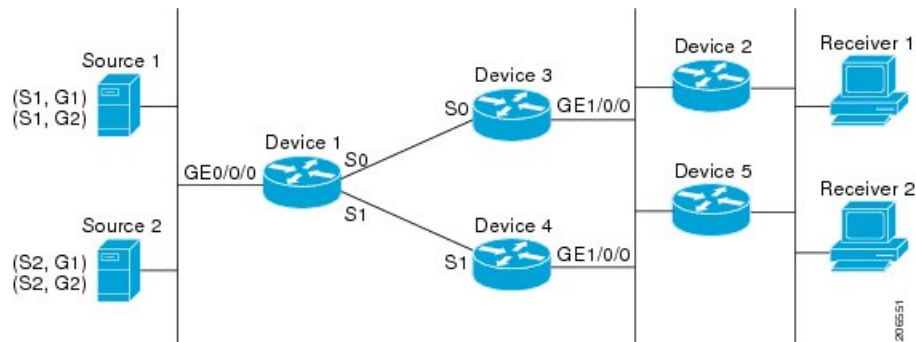
There are also cases where ECMP multicast load splitting with the **ip multicast multipath** command can become ineffective due to the PIM assert process taking over, even when using PIM-SM with (*, G) or (S, G) forwarding or PIM-SSM with (S, G) forwarding.

The figure illustrates a sample topology that is used in this section to explain the effect of ECMP multicast load splitting on the PIM assert process in PIM-SM and PIM-SSM.



Note Although the following illustration and example uses routers in the configuration, any device (router or controller) can be used.

Figure 21: ECMP Multicast Load Splitting and the PIM Assert Process in PIM-SM and PIM-SSM



In the topology illustrated in the figure, if both Device 2 and Device 5 are Cisco devices and are consistently configured for ECMP multicast load splitting with the **ip multicast multipath** command, then load splitting would continue to work as expected; that is, both devices would have Device 3 and Device 4 as equal-cost next hops and would sort the list of equal-cost paths in the same way (by IP address). When applying the multipath hash function, for each (S, G) or (*, G) state, they would choose the same RPF neighbor (either Device 3 or Device 4) and send their PIM joins to this neighbor.

If Device 5 and Device 2 are inconsistently configured with the **ip multicast multipath** command, or if Device 5 is a third-party device, then Device 2 and Device 5 may choose different RPF neighbors for some (*, G) or (S, G) states. For example Device 2 could choose Device 3 for a particular (S, G) state or Device 5 could choose Device 4 for a particular (S, G) state. In this scenario, Device 3 and Device 4 would both start to forward traffic for that state onto Gigabit Ethernet interface 1/0/0, see each other's forwarded traffic, and--to avoid traffic duplication--start the assert process. As a result, for that (S, G) state, the device with the higher IP address for Gigabit Ethernet interface 1/0/0 would forward the traffic. However, both Device 2 and Device

5 would be tracking the winner of the assert election and would send their PIM joins for that state to this assert winner, even if this assert winner is not the same device as the one that they calculated in their RPF selection. For PIM-SM and PIM-SSM, therefore, the operation of ECMP multicast load splitting can only be guaranteed when all downstream devices on a LAN are consistently configured Cisco devices.

ECMP Multicast Load Splitting and Reconvergence When Unicast Routing Changes

When unicast routing changes, all IP multicast routing states reconverge immediately based on the available unicast routing information. Specifically, if one path goes down, the remaining paths reconverge immediately, and if the path comes up again, multicast forwarding will subsequently reconverge to the same RPF paths that were used before the path failed. Reconvergence occurs whether load splitting of IP multicast traffic over ECMP is configured or not.

Use of ECMP Multicast Load Splitting with Static Mroutes

If it is not possible to use an IGP to install equal cost routes for certain sources or RPs, static routes can be configured to specify the equal-cost paths for load splitting. You cannot use static mroutes to configure equal-cost paths because the software does not support the configuration of one static mroute per prefix. There are some workarounds for this limitation using recursive route lookups but the workarounds cannot be applied to equal-cost multipath routing.



Note For more information about configuring static mroutes, see the *Configuring Multiple Static Mroutes in Cisco IOS* configuration note on the Cisco IOS IP multicast FTP site at <ftp://ftpeng.cisco.com/ipmulticast/config-notes/static-mroutes.txt>.

You can specify only static mroutes for equal-cost multipaths in IPv4 multicast; however, those static mroutes would only apply to multicast, or you can specify that the equal-cost multipaths apply to both unicast and multicast routing. In IPv6 multicast, there is no such restriction. Equal-cost multipath mroutes can be configured for static IPv6 mroutes that apply to only unicast routing, only multicast routing, or both unicast and multicast routing.

Alternative Methods of Load Splitting IP Multicast Traffic

Load splitting of IP multicast traffic can also be achieved by consolidating multiple parallel links into a single tunnel over which the multicast traffic is then routed. This method of load splitting is more complex to configure than ECMP multicast load splitting. One such case where configuring load splitting across equal-cost paths using GRE links can be beneficial is the case where the total number of (S, G) or (*, G) states is so small and the bandwidth carried by each state so variable that even the manual engineering of the source or RP addresses cannot guarantee the appropriate load splitting of the traffic.



Note With the availability of ECMP multicast load splitting, tunnels typically only need to be used if per-packet load sharing is required.

IP multicast traffic can also be used to load split across bundle interfaces, such as Fast or Gigabit EtherChannel interfaces, MLPPP link bundles or Multilink Frame Relay (FRF.16) bundles. GRE or other type of tunnels can also constitute such forms of Layer 2 link bundles. Before using such an Layer 2 mechanism, it is necessary to understand how unicast and multicast traffic is load split.

Before load splitting IP multicast traffic across equal-cost paths over a tunnel, you must configure CEF per-packet load balancing or else the GRE packets will not be load balanced per packet.

How to Load Split IP Multicast Traffic over ECMP

Enabling ECMP Multicast Load Splitting

Perform the following tasks to load split IP multicast traffic across multiple equal-cost paths, based on source address.

If two or more equal-cost paths from a source are available, unicast traffic will be load split across those paths. However, by default, multicast traffic is not load split across multiple equal-cost paths. In general, multicast traffic flows down from the RPF neighbor. According to PIM specifications, this neighbor must have the highest IP address if more than one neighbor has the same metric.

Configuring load splitting with the **ip multicast multipath** command causes the system to load split multicast traffic across multiple equal-cost paths based on source address using the S-hash algorithm. When the **ip multicast multipath** command is configured and multiple equal-cost paths exist, the path in which multicast traffic will travel is selected based on the source IP address. Multicast traffic from different sources will be load split across the different equal-cost paths. Load splitting will not occur across equal-cost paths for multicast traffic from the same source sent to different multicast groups.



Note The **ip multicast multipath** command load splits the traffic and does not load balance the traffic. Traffic from a source will use only one path, even if the traffic far outweighs traffic from other sources.

Prerequisites for IP Multicast Load Splitting - ECMP

- You must have an adequate number of sources (at least more than two sources) to enable ECMP multicast load splitting based on source address.
- You must have multiple paths available to the RP to configure ECMP multicast load splitting.



Note Use the **show ip route** command with either the IP address of the source for the *ip-address* argument or the IP address of the RP to validate that there are multiple paths available to the source or RP, respectively. If you do not see multiple paths in the output of the command, you will not be able to configure ECMP multicast load splitting.

- When using PIM-SM with shortest path tree (SPT) forwarding, the T-bit must be set for the forwarding of all (S, G) states.
- Before configuring ECMP multicast load splitting, it is best practice to use the **show ip rpf** command to validate whether sources can take advantage of IP multicast multipath capabilities.

Restrictions for IP Multicast Load Splitting -ECMP

- If two or more equal-cost paths from a source are available, unicast traffic will be load split across those paths. However, by default, multicast traffic is not load split across multiple equal-cost paths. In general, multicast traffic flows down from the RPF neighbor. According to PIM specifications, this neighbor must have the highest IP address if more than one neighbor has the same metric.
- The **ip multicast multipath** command does not support configurations in which the same PIM neighbor IP address is reachable through multiple equal-cost paths. This situation typically occurs if unnumbered interfaces are used. Use different IP addresses for all interfaces when configuring the **ip multicast multipath** command.
- The **ip multicast multipath** command load splits the traffic and does not load balance the traffic. Traffic from a source will use only one path, even if the traffic far outweighs traffic from other sources.

Enabling ECMP Multicast Load Splitting Based on Source Address

Perform this task to enable ECMP multicast load splitting of multicast traffic based on source address (using the S-hash algorithm) to take advantage of multiple paths through the network. The S-hash algorithm is predictable because no randomization is used in calculating the hash value. The S-hash algorithm, however, is subject to polarization because for a given source, the same hash is always picked irrespective of the device on which the hash is being calculated.



Note Enable ECMP multicast load splitting on the device that is to be the receiver for traffic from more than one incoming interfaces, which is opposite to unicast routing. From the perspective of unicast, multicast is active on the sending device connecting to more than one outgoing interfaces.

Before you begin

- You must have an adequate number of sources (at least more than two sources) to enable ECMP multicast load splitting based on source address.
- You must have multiple paths available to the RP to configure ECMP multicast load splitting.



Note Use the **show ip route** command with either the IP address of the source for the *ip-address* argument or the IP address of the RP to validate that there are multiple paths available to the source or RP, respectively. If you do not see multiple paths in the output of the command, you will not be able to configure ECMP multicast load splitting.

- When using PIM-SM with shortest path tree (SPT) forwarding, the T-bit must be set for the forwarding of all (S, G) states.
- Before configuring ECMP multicast load splitting, it is best practice to use the **show ip rpf** command to validate whether sources can take advantage of IP multicast multipath capabilities.

Procedure

| | Command or Action | Purpose |
|---------------|---|--|
| Step 1 | enable Example: <pre>Device> enable</pre> | Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted. |
| Step 2 | configure terminal Example: <pre>Device# configure terminal</pre> | Enters global configuration mode. |
| Step 3 | ip multicast multipath Example: <pre>Device(config)# ip multicast multipath</pre> | Enables ECMP multicast load splitting based on source address using the S-hash algorithm. <ul style="list-style-type: none"> • Because this command changes the way an RPF neighbor is selected, it must be configured consistently on all devices in a redundant topology to avoid looping. • This command does not support configurations in which the same PIM neighbor IP address is reachable through multiple equal-cost paths. This situation typically occurs if unnumbered interfaces are used. Use a different IP address for each interface in a device on which this command is to be configured. • This command load splits the traffic and does not load balance the traffic. Traffic from a source will use only one path, even if the traffic far outweighs traffic from other sources. |
| Step 4 | Repeat step 3 on all the devices in a redundant topology. | -- |
| Step 5 | exit Example: <pre>Device(config)# exit</pre> | Exits global configuration mode and returns to privileged EXEC mode. |
| Step 6 | show ip rpf <i>source-address</i> [<i>group-address</i>] Example: <pre>Device# show ip rpf 10.1.1.2</pre> | (Optional) Displays the information that IP multicast routing uses to perform the RPF check. <ul style="list-style-type: none"> • Use this command to verify RPF selection so as to ensure that IP multicast traffic is being properly load split. |

| | Command or Action | Purpose |
|---------------|--|---|
| Step 7 | show ip route <i>ip-address</i> Example: <pre>Device# show ip route 10.1.1.2</pre> | (Optional) Displays the current state of the IP routing table. <ul style="list-style-type: none"> • Use this command to verify that there are multiple paths available to a source or RP for ECMP multicast load splitting. • For the <i>ip-address</i> argument, enter the IP address of a source to validate that there are multiple paths available to the source (for shortest path trees) or the IP address of an RP to validate that there are multiple paths available to the RP (for shared trees). |

Enabling ECMP Multicast Load Splitting Based on Source and Group Address

Perform this task to enable ECMP multicast load splitting of multicast traffic based on source and group address (using the basic S-G-hash algorithm) to take advantage of multiple paths through the network. The basic S-G-hash algorithm is predictable because no randomization is used in calculating the hash value. The basic S-G-hash algorithm, however, is subject to polarization because for a given source and group, the same hash is always picked irrespective of the device on which the hash is being calculated.

The basic S-G-hash algorithm provides more flexible support for ECMP multicast load splitting than the S-hash algorithm. Using the basic S-G-hash algorithm for load splitting, in particular, enables multicast traffic from devices that send many streams to groups or that broadcast many channels, such as IPTV servers or MPEG video servers, to be more effectively load split across equal-cost paths.



Note Enable ECMP multicast load splitting on the device that is to be the receiver for traffic from more than one incoming interfaces, which is opposite to unicast routing. From the perspective of unicast, multicast is active on the sending device connecting to more than one outgoing interfaces.

Procedure

| | Command or Action | Purpose |
|---------------|---|--|
| Step 1 | enable Example: <pre>Device> enable</pre> | Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted. |
| Step 2 | configure terminal Example: <pre>Device# configure terminal</pre> | Enters global configuration mode. |

| | Command or Action | Purpose |
|---------------|---|---|
| Step 3 | ip multicast multipath s-g-hash basic Example: <pre>Device(config)# ip multicast multipath s-g-hash basic</pre> | Enables ECMP multicast load splitting based on source and group address using the basic S-G-hash algorithm. <ul style="list-style-type: none"> Because this command changes the way an RPF neighbor is selected, it must be configured consistently on all devices in a redundant topology to avoid looping. |
| Step 4 | Repeat Step 3 on all the devices in a redundant topology. | -- |
| Step 5 | exit Example: <pre>Device(config)# exit</pre> | Exits global configuration mode and returns to privileged EXEC mode. |
| Step 6 | show ip rpf source-address [group-address] Example: <pre>Device# show ip rpf 10.1.1.2</pre> | (Optional) Displays the information that IP multicast routing uses to perform the RPF check. <ul style="list-style-type: none"> Use this command to verify RPF selection so as to ensure that IP multicast traffic is being properly load split. |
| Step 7 | show ip route ip-address Example: <pre>Device# show ip route 10.1.1.2</pre> | (Optional) Displays the current state of the IP routing table. <ul style="list-style-type: none"> Use this command to verify that there are multiple paths available to a source or RP for ECMP multicast load splitting. For the <i>ip-address</i> argument, enter the IP address of a source to validate that there are multiple paths available to the source (for shortest path trees) or the IP address of an RP to validate that there are multiple paths available to the RP (for shared trees). |

Enabling ECMP Multicast Load Splitting Based on Source Group and Next-Hop Address

Perform this task to enable ECMP multicast load splitting of multicast traffic based on source, group, and next-hop address (using the next-hop-based S-G-hash algorithm) to take advantage of multiple paths through the network. The next-hop-based S-G-hash algorithm is predictable because no randomization is used in calculating the hash value. Unlike the S-hash and basic S-G-hash algorithms, the hash mechanism used by the next-hop-based S-G-hash algorithm is not subject to polarization.

The next-hop-based S-G-hash algorithm provides more flexible support for ECMP multicast load splitting than S-hash algorithm and eliminates the polarization problem. Using the next-hop-based S-G-hash algorithm for ECMP multicast load splitting enables multicast traffic from devices that send many streams to groups or

that broadcast many channels, such as IPTV servers or MPEG video servers, to be more effectively load split across equal-cost paths.

Procedure

| | Command or Action | Purpose |
|---------------|--|--|
| Step 1 | enable Example: Device> enable | Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted. |
| Step 2 | configure terminal Example: Device# configure terminal | Enters global configuration mode. |
| Step 3 | ip multicast multipath s-g-hash next-hop-based Example: Device(config)# ip multicast multipath s-g-hash next-hop-based | Enables ECMP multicast load splitting based on source, group, and next-hop-address using the next-hop-based S-G-hash algorithm. <ul style="list-style-type: none"> • Because this command changes the way an RPF neighbor is selected, it must be configured consistently on all routers in a redundant topology to avoid looping. <p>Note Be sure to enable the ip multicast multipath command on the router that is supposed to be the receiver for traffic from more than one incoming interfaces, which is opposite to unicast routing. From the perspective of unicast, multicast is active on the sending router connecting to more than one outgoing interfaces.</p> |
| Step 4 | Repeat Steps 1 through 3 on all the routers in a redundant topology. | -- |
| Step 5 | end Example: Device(config)# end | Exits global configuration mode and returns to privileged EXEC mode. |
| Step 6 | show ip rpf source-address [group-address] Example: Device# show ip rpf 10.1.1.2 | (Optional) Displays the information that IP multicast routing uses to perform the RPF check. <ul style="list-style-type: none"> • Use this command to verify RPF selection so as to ensure that IP multicast traffic is being properly load split. |

| | Command or Action | Purpose |
|---------------|--|---|
| Step 7 | show ip route <i>ip-address</i> Example: <pre>Device# show ip route 10.1.1.2</pre> | (Optional) Displays the current state of the IP routing table. <ul style="list-style-type: none"> • Use this command to verify that there are multiple paths available to a source or RP for ECMP multicast load splitting. • For the <i>ip-address</i> argument, enter the IP address of a source to validate that there are multiple paths available to the source (for shortest path trees) or the IP address of an RP to validate that there are multiple paths available to the RP (for shared trees). |

Configuration Examples for Load Splitting IP Multicast Traffic over ECMP

Example Enabling ECMP Multicast Load Splitting Based on Source Address

The following example shows how to enable ECMP multicast load splitting on a router based on source address using the S-hash algorithm:

```
ip multicast multipath
```

Example Enabling ECMP Multicast Load Splitting Based on Source and Group Address

The following example shows how to enable ECMP multicast load splitting on a router based on source and group address using the basic S-G-hash algorithm:

```
ip multicast multipath s-g-hash basic
```

Example Enabling ECMP Multicast Load Splitting Based on Source Group and Next-Hop Address

The following example shows how to enable ECMP multicast load splitting on a router based on source, group, and next-hop address using the next-hop-based S-G-hash algorithm:

```
ip multicast multipath s-g-hash next-hop-based
```

Additional References for IP Multicast Optimization: IP Multicast Load Splitting across Equal-Cost Paths

Related Documents

| Related Topic | Document Title |
|--|---|
| For complete syntax and usage information for the commands used in this chapter. | See the IP Multicast Routing Commands section of the <i>Command Reference (Catalyst 9200 Series Switches)</i> |

Standards and RFCs

| Standard/RFC | Title |
|-----------------|---|
| <i>RFC 4601</i> | Protocol Independent Multicast-Sparse Mode (PIM-SM): Protocol Specification |

Feature History for IP Multicast Optimization: IP Multicast Load Splitting across Equal-Cost Paths

This table provides release and related information for the features explained in this module.

These features are available in all the releases subsequent to the one they were introduced in, unless noted otherwise.

| Release | Feature | Feature Information |
|--------------------------|--|---|
| Cisco IOS XE Fuji 16.9.2 | IP Multicast Optimization: Load Splitting IP Multicast Traffic over ECMP | Load splitting and load balancing are not the same. Load splitting provides a means to randomly distribute (*, G) and (S, G) traffic streams across multiple equal-cost reverse path forwarding (RPF) paths, which does not necessarily result in a balanced IP multicast traffic load on those equal-cost RPF paths. |

| Release | Feature | Feature Information |
|-------------------------------|--|---|
| Cisco IOS XE Cupertino 17.9.1 | IP Multicast Optimization: Load Splitting IP Multicast Traffic over ECMP | This feature was implemented on C9200CX-12P-2X2G, C9200CX-8P-2X2G, and C9200CX-12T-2X2G models of the Cisco Catalyst 9200CX Series Switches, which were introduced in this release. |

Use the Cisco Feature Navigator to find information about platform and software image support. To access Cisco Feature Navigator, go to <https://cfng.cisco.com/>



CHAPTER 15

IP Multicast Optimization: SSM Channel Based Filtering for Multicast

- [Prerequisites for SSM Channel Based Filtering for Multicast Boundaries, on page 275](#)
- [Information About the SSM Channel Based Filtering for Multicast Boundaries, on page 275](#)
- [How to Configure SSM Channel Based Filtering for Multicast Boundaries, on page 276](#)
- [Configuration Examples for SSM Channel Based Filtering for Multicast Boundaries, on page 277](#)
- [Additional References for IP Multicast Optimization: SSM Channel-Based Filtering for Multicast, on page 278](#)
- [Feature History for IP Multicast Optimization: SSM Channel Based Filtering for Multicast, on page 279](#)

Prerequisites for SSM Channel Based Filtering for Multicast Boundaries

IP multicast is enabled on the device using the tasks described in the "Configuring Basic IP Multicast" module of the *IP Multicast: PIM Configuration Guide*.

Information About the SSM Channel Based Filtering for Multicast Boundaries

This section provides information about the SSM channel based filtering for multicast boundaries feature.

Rules for Multicast Boundaries

The SSM Channel Based Filtering for Multicast Boundaries feature expands the **ip multicast boundary** command for control plane filtering support. More than one **ip multicast boundary** command can be applied to an interface.

The following rules govern the **ip multicast boundary** command:

- One instance of the **in** and **out** keywords can be configured on an interface.
- The **in** and **out** keywords can be used for standard or extended access lists.

- Only standard access lists are permitted with the use of the **filter-autorp** keyword or no keyword.
- A maximum of three instances of a command will be allowed on an interface: one instance of **in**, one instance of **out**, and one instance of **filter-autorp** or no keyword.
- When multiple instances of the command are used, the filtering will be cumulative. If a boundary statement with no keyword exists with a boundary statement with the **in** keyword, both access lists will be applied on the in direction and a match on either one will be sufficient.
- All instances of the command apply to both control and data plane traffic.
- Protocol information on the extended access list is parsed to allow reuse and filtering for consistency. An (S,G) operation will be filtered by an extended access list under all conditions stated above for keywords if the access list filters (S,G) traffic for all protocols.

Benefits of SSM Channel Based Filtering for Multicast Boundaries

- This feature allows input on the source interface.
- The access control capabilities are the same for SSM and Any Source Multicast (ASM).

How to Configure SSM Channel Based Filtering for Multicast Boundaries

This section provides steps for configuring SSM channel based filtering for multicast boundaries.

Configuring Multicast Boundaries

Procedure

| | Command or Action | Purpose |
|---------------|--|--|
| Step 1 | enable Example: Device> enable | Enables privileged EXEC mode. |
| Step 2 | configure terminal Example: Device# configure terminal | Enters global configuration mode. |
| Step 3 | ip access-list {standard extended} <i>access-list-name</i> Example: Device(config)# ip access-list 101 | Configures the standard or extended access list. |

| | Command or Action | Purpose |
|---------------|---|--|
| Step 4 | permit <i>protocol</i> host <i>address</i> host <i>address</i> Example: Device(config-ext-nacl)# permit ip host 181.1.2.201 host 232.1.1.11 | Permits specified ip host traffic. |
| Step 5 | deny <i>protocol</i> host <i>address</i> host <i>address</i> Example: Device(config-acl-nacl)# deny ip host 181.1.2.203 host 232.1.1.1 | Denies specified multicast ip group and source traffic. |
| Step 6 | Repeat Step 4 or Step 5 as needed. | Permits and denies specified host and source traffic. |
| Step 7 | interface <i>type</i> interface-number <i>port</i> <i>-number</i> Example: Device(config)# interface gigabitethernet 2/3/0 | Enables interface configuration mode. |
| Step 8 | ip multicast boundary <i>access-list-name</i> [in out filter-autorp] Example: Device(config-if)# ip multicast boundary acc_grpl out | Configures the multicast boundary. Note The filter-autorp keyword does not support extended access lists. |

Configuration Examples for SSM Channel Based Filtering for Multicast Boundaries

This section provides configuration examples of SSM Channel Based filtering for multicast boundaries.

Configuring the Multicast Boundaries Permitting and Denying Traffic Example

The following example permits outgoing traffic for (181.1.2.201, 232.1.1.1) and (181.1.2.202, 232.1.1.1) and denies all other (S,G)s.

```
configure terminal
ip access-list extended acc_grpl
permit ip host 0.0.0.0 232.1.1.1 0.0.0.255
permit ip host 181.1.2.201 host 232.1.1.1
permit udp host 181.1.2.202 host 232.1.1.1
permit ip host 181.1.2.202 host 232.1.1.1
```

```
deny igmp host 181.2.3.303 host 232.1.1.1
interface gigabitethernet 1/0/1
ip multicast boundary acc_grp1 out
```

Configuring the Multicast Boundaries Permitting Traffic Example

The following example permits outgoing traffic for (192.168.2.201, 232.1.1.5) and 192.168.2.202, 232.1.1.5).

```
configure terminal
ip access-list extended acc_grp6
permit ip host 0.0.0.0 232.1.1.1 5.0.0.255
deny udp host 192.168.2.201 host 232.1.1.5
permit ip host 192.168.2.201 host 232.1.1.5
deny pim host 192.168.2.201 host 232.1.1.5
permit ip host 192.168.2.202 host 232.1.1.5
deny igmp host 192.2.3.303 host 232.1.1.1
interface gigabitethernet 1/0/1
ip multicast boundary acc_grp6 out
```

Configuring the Multicast Boundaries Denying Traffic Example

The following example denies a group-range that is announced by the candidate RP. Because the group range is denied, no pim auto-rp mappings are created.

```
configure terminal
ip access-list standard acc_grp10
deny 225.0.0.0 0.255.255.255
permit any
access-list extended acc_grp12
permit pim host 181.1.2.201 host 232.1.1.8
deny udp host 181.1.2.201 host 232.1.1.8
permit pim host 181.1.2.203 0.0.0.255 host 227.7.7.7
permit ip host 0.0.0.0 host 227.7.7.7
permit ip 181.1.2.203 0.0.0.255 host 227.7.7.7
permit ip host 181.1.2.201 host 232.1.1.7
ip access-list extended acc_grp13
deny ip host 181.1.2.201 host 232.1.1.8
permit ip any any
interface gigabitethernet 1/0/1
ip multicast boundary acc_grp10 filter-autorp
ip multicast boundary acc_grp12 out
ip multicast boundary acc_grp13 in
```

Additional References for IP Multicast Optimization: SSM Channel-Based Filtering for Multicast

Related Documents

| Related Topic | Document Title |
|--|---|
| For complete syntax and usage information for the commands used in this chapter. | See the IP Multicast Routing Commands section of the <i>Command Reference (Catalyst 9200 Series Switches)</i> . |

Feature History for IP Multicast Optimization: SSM Channel Based Filtering for Multicast

This table provides release and related information for the features explained in this module.

These features are available in all the releases subsequent to the one they were introduced in, unless noted otherwise.

| Release | Feature | Feature Information |
|-------------------------------|--|---|
| Cisco IOS XE Fuji 16.9.2 | IP Multicast Optimization: SSM Channel Based Filtering for Multicast | The SSM Channel Based Filtering for Multicast Boundaries feature expands the ip multicast boundary command for control plane filtering support. More than one ip multicast boundary command can be applied to an interface. |
| Cisco IOS XE Cupertino 17.9.1 | IP Multicast Optimization: SSM Channel Based Filtering for Multicast | This feature was implemented on C9200CX-12P-2X2G, C9200CX-8P-2X2G, and C9200CX-12T-2X2G models of the Cisco Catalyst 9200CX Series Switches, which were introduced in this release. |

Use the Cisco Feature Navigator to find information about platform and software image support. To access Cisco Feature Navigator, go to <https://cfng.cisco.com/>



CHAPTER 16

IP Multicast Optimization: IGMP State Limit

- [Prerequisites for IGMP State Limit, on page 281](#)
- [Restrictions for IGMP State Limit, on page 281](#)
- [Information About IGMP State Limit, on page 281](#)
- [How to Configure IGMP State Limit, on page 283](#)
- [Configuration examples for IGMP State Limit, on page 284](#)
- [Additional References, on page 286](#)
- [Feature History for IP Multicast Optimization: IGMP State Limit, on page 286](#)

Prerequisites for IGMP State Limit

- IP multicast is enabled and the Protocol Independent Multicast (PIM) interfaces are configured using the tasks described in the "Configuring Basic IP Multicast" module of the *IP Multicast: PIM Configuration Guide*.
- ALL ACLs must be configured. For information, see the "Creating an IP Access List and Applying It to an Interface" module of the *Security Configuration Guide: Access Control Lists* guide.

Restrictions for IGMP State Limit

You can configure only one global limit per device and one limit per interface.

Information About IGMP State Limit

This section provides information about IGMP state limit.

IGMP State Limit

The IGMP State Limit feature allows for the configuration of IGMP state limiters, which impose limits on mroute states resulting from IGMP membership reports (IGMP joins) on a global or per interface basis. Membership reports exceeding the configured limits are not entered into the IGMP cache. This feature can be used to prevent DoS attacks or to provide a multicast CAC mechanism in network environments where all the multicast flows roughly utilize the same amount of bandwidth.



Note IGMP state limiters impose limits on the number of mroute states resulting from IGMP, IGMP v3lite, and URL Rendezvous Directory (URD) membership reports on a global or per interface basis.

IGMP State Limit Feature Design

- Configuring IGMP state limiters in global configuration mode specifies a global limit on the number of IGMP membership reports that can be cached.
- Configuring IGMP state limiters in interface configuration mode specifies a limit on the number of IGMP membership reports on a per interface basis.
- Use ACLs to prevent groups or channels from being counted against the interface limit. A standard or an extended ACL can be specified. A standard ACL can be used to define the (*, G) state to be excluded from the limit on an interface. An extended ACLs can be used to define the (S, G) state to be excluded from the limit on an interface. An extended ACL also can be used to define the (*, G) state to be excluded from the limit on an interface, by specifying 0.0.0.0 for the source address and source wildcard--referred to as (0, G)--in the permit or deny statements that compose the extended access list.
- You can only configure one global limit per device and one limit per interface.

Mechanics of IGMP State Limiters

The mechanics of IGMP state limiters are as follows:

- Each time a router receives an IGMP membership report for a particular group or channel, the Cisco IOS software checks to see if either the limit for the global IGMP state limiter or the limit for the per interface IGMP state limiter has been reached.
- If only a global IGMP state limiter has been configured and the limit has not been reached, IGMP membership reports are honored. When the configured limit has been reached, subsequent IGMP membership reports are then ignored (dropped) and a warning message in one of the following formats is generated:
 - ```
%IGMP-6-IGMP_GROUP_LIMIT: IGMP limit exceeded for <group (*, group address)> on <interface type number> by host <ip address>
```
  - ```
%IGMP-6-IGMP_CHANNEL_LIMIT: IGMP limit exceeded for <channel (source address, group address)> on <interface type number> by host <ip address>
```
- If only per interface IGMP state limiters are configured, then each limit is only counted against the interface on which it was configured.
- If both a global IGMP state limiter and per interface IGMP state limiters are configured, the limits configured for the per interface IGMP state limiters are still enforced but are constrained by the global limit.

How to Configure IGMP State Limit

This section describes how to configure IGMP state limit.

Configuring IGMP State Limiters

IGMP state limiters impose limits on the number of mroute states resulting from IGMP, IGMP v3lite, and URD membership reports on a global or per interface basis.

Configuring Global IGMP State Limiters

Perform this optional task to configure one global IGMP state limiter per device.

Procedure

| | Command or Action | Purpose |
|---------------|--|---|
| Step 1 | enable Example: Device> enable | Enables privileged EXEC mode. • Enter your password if prompted. |
| Step 2 | configure terminal Example: Device# configure terminal | Enters global configuration mode. |
| Step 3 | ip igmp limit <i>number</i> Example: Device(config)# ip igmp limit 150 | Configures a global limit on the number of mroute states resulting from IGMP membership reports (IGMP joins). |
| Step 4 | end Example: Device(config-if)# end | Ends the current configuration session and returns to privileged EXEC mode. |
| Step 5 | show ip igmp groups Example: Device# show ip igmp groups | (Optional) Displays the multicast groups with receivers that are directly connected to the device and that were learned through IGMP. |

Configuring Per Interface IGMP State Limiters

Perform this optional task to configure a per interface IGMP state limiter.

Procedure

| | Command or Action | Purpose |
|---------------|---|--|
| Step 1 | enable Example: Device> enable | Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted. |
| Step 2 | configure terminal Example: Device# configure terminal | Enters global configuration mode. |
| Step 3 | interface <i>type number</i> Example: Device(config)# interface GigabitEthernet 1/0/0 | Enters interface configuration mode. <ul style="list-style-type: none"> • Specify an interface that is connected to hosts. |
| Step 4 | ip igmp limit <i>number</i> [except <i>access-list</i>] Example: Device(config-if)# ip igmp limit 100 | Configures a per interface limit on the number of mroutes states created as a result of IGMP membership reports (IGMP joins). |
| Step 5 | Do one of the following: <ul style="list-style-type: none"> • exit • end Example: Device(config-if)# exit Device(config-if)# end | <ul style="list-style-type: none"> • (Optional) Ends the current configuration session and returns to global configuration mode. Repeat steps 3 and 4 to configure a per interface limiter on another interface. • Ends the current configuration session and returns to privileged EXEC mode. |
| Step 6 | show ip igmp interface [<i>type number</i>] Example: Device# show ip igmp interface | (Optional) Displays information about the status and configuration of IGMP and multicast routing on interfaces. |
| Step 7 | show ip igmp groups Example: Device# show ip igmp groups | (Optional) Displays the multicast groups with receivers that are directly connected to the device and that were learned through IGMP. |

Configuration examples for IGMP State Limit

This section show configuration examples of IGMP state limit.

Configuring IGMP State Limiters Example

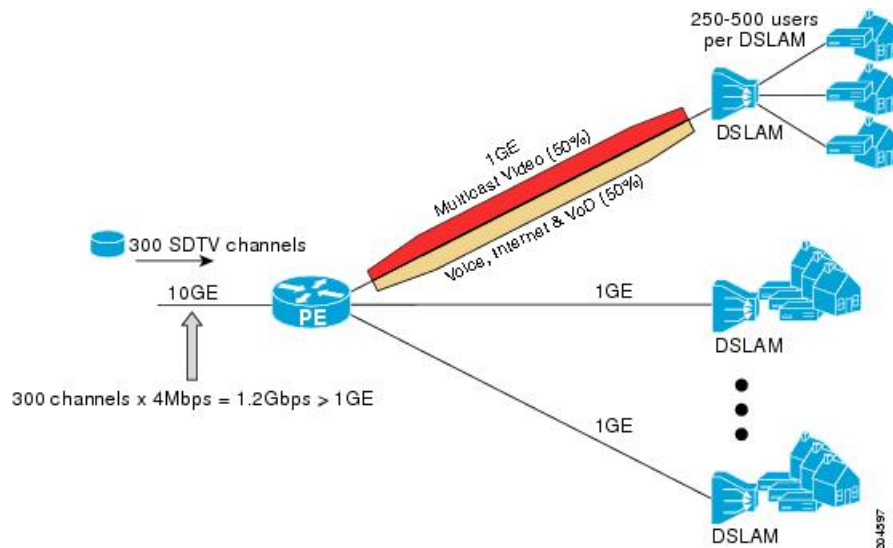
The following example shows how to configure IGMP state limiters to provide multicast CAC in a network environment where all the multicast flows roughly utilize the same amount of bandwidth.

This example uses the topology illustrated in the figure.



Note Although the following illustration and example uses routers in the configuration, any device (router or switch) can be used.

Figure 22: IGMP State Limit Example Topology



In this example, a service provider is offering 300 Standard Definition (SD) TV channels. Each SD channel utilizes approximately 4 Mbps.

The service provider must provision the Gigabit Ethernet interfaces on the PE router connected to the Digital Subscriber Line Access Multiplexers (DSLAMs) as follows: 50% of the link's bandwidth (500 Mbps) must be available to subscribers of the Internet, voice, and video on demand (VoD) service offerings while the remaining 50% (500 Mbps) of the link's bandwidth must be available to subscribers of the SD channel offerings.

Because each SD channel utilizes the same amount of bandwidth (4 Mbps), per interface IGMP state limiters can be used to provide the necessary CAC to provision the services being offered by the service provider. To determine the required CAC needed per interface, the total number of channels is divided by 4 (because each channel utilizes 4 Mbps of bandwidth). The required CAC needed per interface, therefore, is as follows:

$$500\text{Mbps} / 4\text{Mbps} = 125 \text{ mroutes}$$

Once the required CAC is determined, the service provider uses the results to configure the per IGMP state limiters required to provision the Gigabit Ethernet interfaces on the PE router. Based on the network's CAC requirements, the service provider must limit the SD channels that can be transmitted out a Gigabit Ethernet interface (at any given time) to 125. Configuring a per interface IGMP state limit of 125 for the SD channels provisions the interface for 500 Mbps of bandwidth, the 50% of the link's bandwidth that must always be available (but never exceeded) for the SD channel offerings.

The following configuration shows how the service provider uses a per interface mroute state limiter to provision interface Gigabit Ethernet 0/0/0 for the SD channels and Internet, Voice, and VoD services being offered to subscribers:

```
interface GigabitEthernet0/0/0
description --- Interface towards the DSLAM ---
.
.
.
ip igmp limit 125
```

Additional References

Related Documents

| Related Topic | Document Title |
|--|---|
| For complete syntax and usage information for the commands used in this chapter. | See the IP Multicast Routing Commands section of the <i>Command Reference (Catalyst 9200 Series Switches)</i> |

Feature History for IP Multicast Optimization: IGMP State Limit

This table provides release and related information for the features explained in this module.

These features are available in all the releases subsequent to the one they were introduced in, unless noted otherwise.

| Release | Feature | Feature Information |
|--------------------------|---|--|
| Cisco IOS XE Fuji 16.9.2 | IP Multicast Optimization: IGMP State Limit | The IGMP State Limit feature allows for the configuration of IGMP state limiters, which impose limits on mroute states resulting from IGMP membership reports (IGMP joins) on a global or per interface basis. Membership reports exceeding the configured limits are not entered into the IGMP cache. |

| Release | Feature | Feature Information |
|-------------------------------|---|---|
| Cisco IOS XE Cupertino 17.9.1 | IP Multicast Optimization: IGMP State Limit | This feature was implemented on C9200CX-12P-2X2G, C9200CX-8P-2X2G, and C9200CX-12T-2X2G models of the Cisco Catalyst 9200CX Series Switches, which were introduced in this release. |

Use the Cisco Feature Navigator to find information about platform and software image support. To access Cisco Feature Navigator, go to <https://cfng.cisco.com/>



CHAPTER 17

Troubleshooting IP Multicast

- [Overview](#), on page 289
- [Support Articles](#), on page 289
- [Feedback Request](#), on page 290
- [Disclaimer and Caution](#), on page 290

Overview

This chapter provides links to documents authored by Cisco subject matter experts (SMEs). They aim to help you resolve technical issues without requiring a support ticket. If these documents are unable to resolve your issue, we recommend visiting the applicable [Cisco Community](#). There is a wealth of information and advice available from fellow Cisco customers who may have experienced this issue already and provided a solution. If you are not able to find a resolution on the Community, it may be best that you raise a support ticket at [Cisco Support](#). In cases where a support ticket has to be raised, these documents provide guidance about the data that should be collected and added to the support ticket. Specify the support document you referred, and TAC can create an improvement request with the document owner.

Support Articles

The documents in this section were created using specific software and hardware listed in the Components Used section of each article. However, this does not mean that they are limited to what is listed in Components Used, and generally remain relevant for later versions of software and hardware. Note that there could be some changes in the software or hardware that can cause commands to stop working, the syntax to change, or GUIs and CLIs to look different from one release to another.

The following are the support articles associated with this technology:

| Document | Description |
|--|---|
| Secure IP Multicast Deployments | This document describes the best practices to secure an IP multicast network infrastructure. |
| Implement SSDP Best Practices on Catalyst 9000 Series Switches | This document describes best practice configurations designed to drop or limit the Simple Service Discovery Protocol (SSDP) packets on Catalyst 9000 series switches. |

Feedback Request

Your input helps. A key aspect to improving these support documents is customer feedback. Note that these documents are owned and maintained by multiple teams within Cisco. If you find an issue specific to the document (unclear, confusing, information missing, etc):

- Provide feedback using the **Feedback** button located at the right panel of the corresponding article. The document owner will be notified, and will either update the article, or flag it for removal.
- Include information regarding the section, area, or issue you had with the document and what could be improved. Provide as much detail as possible.

Disclaimer and Caution

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is live, ensure that you understand the potential impact of any command.