Notices

The information contained herein is subject to change without notice. The only warranties for Hewlett Packard Enterprise products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. Hewlett Packard Enterprise shall not be liable for technical or editorial errors or omissions contained herein.


Links to third-party websites take you outside the Hewlett Packard Enterprise website. Hewlett Packard Enterprise has no control over and is not responsible for information outside the Hewlett Packard Enterprise website.

Acknowledgments

Intel®, Itanium®, Pentium®, Xeon®, Intel Inside®, and the Intel Inside logo are trademarks of Intel Corporation in the U.S. and other countries.

Microsoft® and Windows® are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.

Adobe® and Acrobat® are trademarks of Adobe Systems Incorporated.

Java® and Oracle® are registered trademarks of Oracle and/or its affiliates.

UNIX® is a registered trademark of The Open Group.
### Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>About this guide</strong></td>
<td>8</td>
</tr>
<tr>
<td>Applicable products</td>
<td>8</td>
</tr>
<tr>
<td>Switch prompts used in this guide</td>
<td>8</td>
</tr>
<tr>
<td>Terminology Change</td>
<td>9</td>
</tr>
<tr>
<td><strong>Multimedia traffic control with IP multicast (IGMP)</strong></td>
<td>10</td>
</tr>
<tr>
<td>Overview of IGMP</td>
<td>10</td>
</tr>
<tr>
<td>Enabling IGMP</td>
<td>10</td>
</tr>
<tr>
<td>Configuring and displaying IGMP (CLI)</td>
<td>10</td>
</tr>
<tr>
<td>ip igmp fastleave</td>
<td>15</td>
</tr>
<tr>
<td>ip igmp forcedfastleave</td>
<td>16</td>
</tr>
<tr>
<td>Configuring IGMP proxy (CLI)</td>
<td>17</td>
</tr>
<tr>
<td>IGMP general operation and features</td>
<td>19</td>
</tr>
<tr>
<td>How IGMP operates</td>
<td>21</td>
</tr>
<tr>
<td>About using the switch as querier</td>
<td>28</td>
</tr>
<tr>
<td>Well-known or reserved multicast addresses excluded from IP multicast filtering</td>
<td>29</td>
</tr>
<tr>
<td>IP multicast filters</td>
<td>30</td>
</tr>
<tr>
<td>IGMPv3</td>
<td>30</td>
</tr>
<tr>
<td><strong>PIM-DM (Dense Mode)</strong></td>
<td>41</td>
</tr>
<tr>
<td>Overview of PIM-DM</td>
<td>41</td>
</tr>
<tr>
<td>Global and PIM configuration contexts</td>
<td>41</td>
</tr>
<tr>
<td>PIM VLAN (interface) configuration context</td>
<td>43</td>
</tr>
<tr>
<td>Displaying PIM data and configuration settings</td>
<td>48</td>
</tr>
<tr>
<td>PIM-DM</td>
<td>57</td>
</tr>
<tr>
<td>PIM-DM operation</td>
<td>58</td>
</tr>
<tr>
<td>Configuring PIM-DM</td>
<td>60</td>
</tr>
<tr>
<td>PIM-DM DT</td>
<td>61</td>
</tr>
<tr>
<td>Operating notes</td>
<td>64</td>
</tr>
<tr>
<td>Troubleshooting</td>
<td>65</td>
</tr>
<tr>
<td>Messages related to PIM operation</td>
<td>66</td>
</tr>
<tr>
<td>Applicable RFCs</td>
<td>68</td>
</tr>
<tr>
<td>Exceptions to Support for RFC 2932 - Multicast Routing MIB</td>
<td>69</td>
</tr>
<tr>
<td><strong>PIM-SM (Sparse Mode)</strong></td>
<td>70</td>
</tr>
<tr>
<td>Configuring router protocol independent multicast (PIM)</td>
<td>70</td>
</tr>
<tr>
<td>Configuring PIM-SM on the router</td>
<td>71</td>
</tr>
<tr>
<td>VLAN context commands for configuring PIM-SM</td>
<td>73</td>
</tr>
<tr>
<td>Router PIM context commands for configuring PIM-SM operation</td>
<td>76</td>
</tr>
<tr>
<td>Configuring C-RPs on PIM-SM routers</td>
<td>78</td>
</tr>
<tr>
<td>Enabling, disabling, or changing router PIM notification traps</td>
<td>80</td>
</tr>
<tr>
<td>Changing the global join-prune interval on the router</td>
<td>81</td>
</tr>
<tr>
<td>Changing the shortest-path tree (SPT) operation</td>
<td>81</td>
</tr>
<tr>
<td>Statically configuring an RP to accept multicast traffic</td>
<td>81</td>
</tr>
<tr>
<td>PIM RPF override configuration</td>
<td>83</td>
</tr>
<tr>
<td>Displaying configured RPF overrides</td>
<td>84</td>
</tr>
<tr>
<td>Displaying PIM route data</td>
<td>85</td>
</tr>
<tr>
<td>Topic</td>
<td>Page</td>
</tr>
<tr>
<td>-----------------------------------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>Viewing PIM-specific data</td>
<td>87</td>
</tr>
<tr>
<td>Displaying PIM neighbor data</td>
<td>89</td>
</tr>
<tr>
<td>Display pending join requests</td>
<td>90</td>
</tr>
<tr>
<td>Displaying BSR data</td>
<td>91</td>
</tr>
<tr>
<td>Displaying the current RP set</td>
<td>92</td>
</tr>
<tr>
<td>Displaying C-RP data</td>
<td>93</td>
</tr>
<tr>
<td>PIM-SM overview</td>
<td>94</td>
</tr>
<tr>
<td>PIM-SM features</td>
<td>95</td>
</tr>
<tr>
<td>PIM-SM operation and router types</td>
<td>96</td>
</tr>
<tr>
<td>PIM-SM DT</td>
<td>98</td>
</tr>
<tr>
<td>PIM-SM router types</td>
<td>98</td>
</tr>
<tr>
<td>Operating rules and recommendations</td>
<td>104</td>
</tr>
<tr>
<td>Configuration steps for PIM-SM</td>
<td>104</td>
</tr>
<tr>
<td>Operating notes</td>
<td>108</td>
</tr>
<tr>
<td>Routing Basics</td>
<td>109</td>
</tr>
<tr>
<td>Overview of IP routing</td>
<td>109</td>
</tr>
<tr>
<td>Viewing the IP route table</td>
<td>109</td>
</tr>
<tr>
<td>Increasing ARP age timeout (CLI)</td>
<td>109</td>
</tr>
<tr>
<td>Setting and viewing the arp-age value</td>
<td>110</td>
</tr>
<tr>
<td>Reconfiguring the router ID (optional)</td>
<td>111</td>
</tr>
<tr>
<td>Enabling proxy ARP</td>
<td>111</td>
</tr>
<tr>
<td>Configuring source MAC based ARP attack detection (ARP throttle)</td>
<td>112</td>
</tr>
<tr>
<td>Enabling forwarding of IP directed broadcasts (CLI)</td>
<td>116</td>
</tr>
<tr>
<td>Disabling replies to broadcast ping requests</td>
<td>121</td>
</tr>
<tr>
<td>IP interfaces</td>
<td>122</td>
</tr>
<tr>
<td>IP tables and caches</td>
<td>122</td>
</tr>
<tr>
<td>IP forwarding cache</td>
<td>124</td>
</tr>
<tr>
<td>IP route exchange protocols</td>
<td>124</td>
</tr>
<tr>
<td>Configuring IP parameters for routing switches</td>
<td>127</td>
</tr>
<tr>
<td>Configuring forwarding parameters</td>
<td>129</td>
</tr>
<tr>
<td>Configuring ICMP</td>
<td>129</td>
</tr>
<tr>
<td>Static Routing</td>
<td>131</td>
</tr>
<tr>
<td>Configuring an IPv4 Route</td>
<td>131</td>
</tr>
<tr>
<td>Configuring an IPv6 Route</td>
<td>132</td>
</tr>
<tr>
<td>Viewing static route information</td>
<td>135</td>
</tr>
<tr>
<td>Static route types</td>
<td>135</td>
</tr>
<tr>
<td>Log violation of Null-Route</td>
<td>138</td>
</tr>
<tr>
<td>Overview of Null-Route</td>
<td>138</td>
</tr>
<tr>
<td>Commands</td>
<td>138</td>
</tr>
<tr>
<td>Modifying existing commands</td>
<td>141</td>
</tr>
<tr>
<td>Show commands enhancement</td>
<td>142</td>
</tr>
<tr>
<td>Restrictions</td>
<td>142</td>
</tr>
<tr>
<td>Routing Information Protocol (RIP)</td>
<td>143</td>
</tr>
<tr>
<td>Overview of RIP</td>
<td>143</td>
</tr>
<tr>
<td>Configuring RIP parameters</td>
<td>143</td>
</tr>
<tr>
<td>Configuring for redistribution</td>
<td>146</td>
</tr>
<tr>
<td>Changing the route loop prevention method</td>
<td>147</td>
</tr>
<tr>
<td>Viewing RIP information</td>
<td>148</td>
</tr>
<tr>
<td>RIP parameters and defaults</td>
<td>152</td>
</tr>
<tr>
<td>Configuring RIP redistribution</td>
<td>153</td>
</tr>
<tr>
<td>Changing the route loop prevention method</td>
<td>154</td>
</tr>
<tr>
<td>RIPv2 MD5 authentication</td>
<td>155</td>
</tr>
<tr>
<td>Page</td>
<td></td>
</tr>
<tr>
<td>------</td>
<td></td>
</tr>
<tr>
<td><strong>RIPng</strong></td>
<td>158</td>
</tr>
<tr>
<td>RIPng for IPv6</td>
<td>158</td>
</tr>
<tr>
<td>Configure RIPng</td>
<td>159</td>
</tr>
<tr>
<td>VLAN Level Configuration</td>
<td>161</td>
</tr>
<tr>
<td>Show commands</td>
<td>162</td>
</tr>
<tr>
<td>Debug commands</td>
<td>165</td>
</tr>
<tr>
<td>Additional commands</td>
<td>165</td>
</tr>
</tbody>
</table>

**Policy Based Routing** | 168 |
Creating a Policy Based Routing Policy | 168 |

**Open Shortest Path First Protocol (OSPF)** | 170 |
Configuring OSPF on the routing switch | 170 |
Assigning the routing switch to OSPF areas | 172 |
Configuring external route redistribution in an OSPF domain (optional) | 177 |
Configuring ranges on an ABR to reduce advertising to the backbone | 179 |
Influencing route choices by changing the administrative distance default | 182 |
Changing OSPF trap generation choices | 182 |
Adjusting performance by changing the VLAN or subnet interface settings | 183 |
Configuring OSPF interface authentication | 186 |
Configuring a virtual link | 187 |
Configuring OSPF authentication on a virtual link | 190 |
Configuring a passive OSPF interface | 191 |
Configuring OSPF Network Type | 192 |
Configuring the calculation interval | 192 |
Viewing OSPF information | 194 |
Enabling load sharing among next-hop routes | 216 |
Overview of OSPF | 217 |
OSPF area types | 221 |
OSPF RFC compliance | 223 |
Reducing AS external LSAs and Type-3 summary LSAs | 223 |
Equal cost multi-path routing (ECMP) | 224 |
Dynamic OSPF activation and configuration | 226 |
Changing the RFC 1583 OSPF compliance setting | 228 |
Assigning the routing switch to OSPF areas | 228 |
Configuring for external route redistribution in an OSPF domain | 228 |
Configuring ranges on an ABR to reduce advertising to the backbone | 229 |
Influencing route choices by changing the administrative distance default (optional) | 230 |
Adjusting performance by changing the VLAN or subnet interface settings | 230 |
Configuring OSPF interface authentication | 230 |
Configuring an ABR to use a virtual link to the backbone | 230 |
Adjusting virtual link performance by changing the interface settings | 231 |
Configuring OSPF authentication on a virtual link | 231 |
About OSPF passive | 231 |
About configuring shortest path first (SPF) scheduling | 231 |
Graceful shutdown of OSPF routing | 232 |
Global OSPF Cost Setting | 233 |

**Route Policy** | 235 |
Using prefix lists | 235 |
Creating a route map | 238 |
Using match commands | 239 |
Using set commands | 241 |
Route policy overview | 242 |
Route maps | 243 |
Match commands | 244 |
Using route policy in route redistribution .......................................................... 244

**ICMP Router Discovery Protocol** ................................................................. 256
   Configuring IRDP ................................................................. 256
   Viewing IRDP information .......................................................... 258

**Dynamic Host Configuration Protocol** ......................................................... 259
   Enabling DHCP relay .......................................................... 259
   Using DHCP Option 12 to send a hostname ................................. 259
   Configuring a BOOTP/DHCP relay gateway ................................. 260
   Configuring an IP helper address ............................................. 261
   Disabling the hop count in DHCP requests ................................. 261
   Verifying the DHCP relay configuration ..................................... 261
   Viewing the MAC address for a routing switch ................................. 263
   Configuring Option 82 .......................................................... 263
   Overview of DHCP ................................................................ 265
   Hop count in DHCP requests .................................................. 266
   DHCP Option 82 ................................................................. 266

**User Datagram Protocol** ............................................................................ 275
   Configuring and enabling UDP broadcast forwarding ...................... 275
   UDP broadcast forwarding ........................................................ 278
   Subnet masking for UDP forwarding addresses .............................. 280

**Virtual Router Redundancy Protocol (VRRP)** ............................................... 281
   VRRP overview ................................................................. 281
   Configuring VRRP .............................................................. 281
   Selecting a Version of VRRP .................................................. 282
   Configuring a VR instance on a VLAN interface ............................ 283
   Changing VR advertisement interval and source IP address ......... 284
   Configuring preempt mode on VRRP backup routers .................... 285
   Enabling or disabling VRRP operation on a VR .............................. 285
   Dynamically changing the priority of the VR ................................. 285
   Removing all tracked entities .................................................... 286
   Viewing VRRP tracked entities .................................................. 287
   Forcing the backup VR operating as main router to relinquish ownership of the VR instance 287
   Forcing the backup VR to take ownership of the VR instance ......... 287
   Configuring the Authentication Data Field ..................................... 287
   Pinging the virtual IP of a backup router ..................................... 288
   Specifying the time a router waits before taking control of the VIP 292
   Viewing VRRP configuration data ................................................. 292
   Using the debug command with the VRRP option ......................... 300
   General operation ................................................................. 300
   Steps for provisioning VRRP operation ........................................ 306
   Dynamically changing the priority of the VR ................................. 310
   Failover operation ............................................................... 310
   Pinging the virtual IP of a backup router ..................................... 310
   Using the Pre-empt Delay Timer (PDT) ......................................... 310
   Standards compliance ............................................................. 312
   Operating notes ................................................................. 312
   VRRP-Services enhancements .................................................. 313
   VRRP-Show command enhancement ......................................... 316
   Error messages—Track interface ............................................... 317

**Border Gateway Protocol (BGP)** ................................................................. 318
   Introduction ................................................................. 318
   Configuring BGP globally ........................................................ 318
# Aruba 3810M/5800R Multicasting and Routing Guide

<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Configuring BGP policy globally</td>
<td>320</td>
</tr>
<tr>
<td>BGP graceful restart</td>
<td>325</td>
</tr>
<tr>
<td>Neighbor configuration and neighbor policy config</td>
<td>326</td>
</tr>
<tr>
<td>Synchronizing BGP-IGP</td>
<td>335</td>
</tr>
<tr>
<td>BGP path attributes</td>
<td>336</td>
</tr>
<tr>
<td>BGP route selection</td>
<td>339</td>
</tr>
<tr>
<td>Protocols and standards</td>
<td>341</td>
</tr>
<tr>
<td>BGP extensions</td>
<td>342</td>
</tr>
<tr>
<td>BGP basic configuration</td>
<td>343</td>
</tr>
<tr>
<td>Displaying information about BGP configuration</td>
<td>349</td>
</tr>
<tr>
<td>BGP configuration examples</td>
<td>352</td>
</tr>
<tr>
<td>BGP show routines</td>
<td>365</td>
</tr>
<tr>
<td>BGP solution use cases</td>
<td>370</td>
</tr>
<tr>
<td>Troubleshooting BGP</td>
<td>373</td>
</tr>
<tr>
<td><strong>Bidirectional Forwarding Detection (BFD)</strong></td>
<td>375</td>
</tr>
<tr>
<td>Commands</td>
<td>375</td>
</tr>
<tr>
<td>Configuration Requirements</td>
<td>384</td>
</tr>
<tr>
<td>BFD static routing</td>
<td>385</td>
</tr>
<tr>
<td>Smart link configuration commands</td>
<td>397</td>
</tr>
<tr>
<td>VXLAN show commands</td>
<td>401</td>
</tr>
<tr>
<td>Prerequisites</td>
<td>403</td>
</tr>
<tr>
<td><strong>Support and other resources</strong></td>
<td>404</td>
</tr>
<tr>
<td>Accessing Aruba Support</td>
<td>404</td>
</tr>
<tr>
<td>Other useful sites</td>
<td>404</td>
</tr>
<tr>
<td>Accessing updates</td>
<td>404</td>
</tr>
<tr>
<td>Warranty information</td>
<td>405</td>
</tr>
<tr>
<td>Regulatory information</td>
<td>405</td>
</tr>
<tr>
<td>Documentation feedback</td>
<td>405</td>
</tr>
<tr>
<td><strong>Apple's Bonjour and Google's Chromecast</strong></td>
<td>406</td>
</tr>
<tr>
<td>Overview of Apple's Bonjour and Google's Chromecast</td>
<td>406</td>
</tr>
<tr>
<td>mDNS Gateway</td>
<td>406</td>
</tr>
<tr>
<td>Limitations of the mDNS gateware and Chromecast</td>
<td>410</td>
</tr>
<tr>
<td>Profile and rule limit details</td>
<td>410</td>
</tr>
<tr>
<td>mdns enable</td>
<td>410</td>
</tr>
<tr>
<td>Create mDNS reflection</td>
<td>410</td>
</tr>
<tr>
<td>Create or delete a mDNS profile</td>
<td>411</td>
</tr>
<tr>
<td>rule</td>
<td>411</td>
</tr>
<tr>
<td>Set the specific mDNS profile for VLAN</td>
<td>412</td>
</tr>
<tr>
<td>Set the global mDNS profile</td>
<td>412</td>
</tr>
<tr>
<td>Show mdns</td>
<td>412</td>
</tr>
<tr>
<td>Show mDNS gateway</td>
<td>413</td>
</tr>
<tr>
<td>Show mDNS profile configuration</td>
<td>413</td>
</tr>
<tr>
<td>Show mDNS profile name</td>
<td>414</td>
</tr>
<tr>
<td>Debug mDNS</td>
<td>414</td>
</tr>
</tbody>
</table>
This guide provides information on how to configure IGMP, PIM and routing protocols.

Applicable products
This guide applies to these products:

Switch prompts used in this guide
Examples in this guide are representative and may not match your particular switch/environment. Examples use simplified prompts as follows:

Table 1: Switch prompts used in this guide

<table>
<thead>
<tr>
<th>Prompt</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>switch#</td>
<td># indicates manager context (authority).</td>
</tr>
<tr>
<td>switch&gt;</td>
<td>&gt; indicates operator context (authority).</td>
</tr>
<tr>
<td>switch(config)#</td>
<td>(config) indicates the config context.</td>
</tr>
<tr>
<td>switch(vlan-x)#</td>
<td>(vlan-x) indicates the vlan context of config, where x represents the VLAN ID. For example: switch(vlan-128)#.</td>
</tr>
<tr>
<td>switch(eth-x)#</td>
<td>(eth-x) indicates the interface context of config, where x represents the interface. For example: switch(eth-48)#.</td>
</tr>
<tr>
<td>switch-Stack#</td>
<td>Stack indicates that stacking is enabled.</td>
</tr>
<tr>
<td>switch-Stack(config)#</td>
<td>Stack(config) indicates the config context while stacking is enabled.</td>
</tr>
<tr>
<td>switch-Stack( stacking)#</td>
<td>Stack(stacking) indicates the stacking context of config while stacking is enabled.</td>
</tr>
<tr>
<td>switch-Stack(vlan-x)#</td>
<td>Stack(vlan-x) indicates the vlan context of config while stacking is enabled, where x represents the VLAN ID. For example: switch-Stack(vlan-128)#.</td>
</tr>
</tbody>
</table>
Terminology Change
As part of advancing HPE’s commitment to racial justice, we are taking a much-needed step in overhauling HPE engineering terminology to reflect our belief system of diversity and inclusion. Some legacy products and publications may continue to include terminology that seemingly evokes bias against specific groups of people. Such content is not representative of our HPE culture and moving forward, Aruba will replace racially insensitive terms and instead use the following new language:

<table>
<thead>
<tr>
<th>Usage</th>
<th>Old Language</th>
<th>New Language</th>
</tr>
</thead>
<tbody>
<tr>
<td>Switch Security</td>
<td>Master</td>
<td>Main</td>
</tr>
<tr>
<td>Switch Routing</td>
<td>Master</td>
<td>Main Router</td>
</tr>
<tr>
<td>Smart Link</td>
<td>Master-Slave</td>
<td>Primary-Secondary</td>
</tr>
<tr>
<td>Chassis Events, IPv6 Configuration, and Troubleshooting</td>
<td>Master-Slave</td>
<td>Management-Slot</td>
</tr>
<tr>
<td>Switch Stack</td>
<td>Master-Slave</td>
<td>Conductor-Member</td>
</tr>
<tr>
<td>Switch Security, Configuration and Routing</td>
<td>Blacklist, Whitelist</td>
<td>Denylist, Allowlist</td>
</tr>
<tr>
<td>Route Type</td>
<td>Blackhole Route</td>
<td>Null Route</td>
</tr>
<tr>
<td>Type of Hackers</td>
<td>Black Hat, White Hat</td>
<td>Unethical, Ethical</td>
</tr>
</tbody>
</table>

Prompt | Explanation
---|-------------------
switch-Stack (eth-x/y)# | Stack(eth-<x/y>) indicates the interface context of config, in the form (eth-<member-in-stack>/<interface>). For example: switch(eth-1/48)#
Overview of IGMP

This chapter describes multimedia traffic control with IP multicast—Internet Group Management Protocol (IGMP). IGMP reduces unnecessary bandwidth usage on a per-port basis.

More information
IGMP general operation and features

Enabling IGMP

IGMP is disabled in the default factory configuration. To enable:

Procedure:

1. If multiple VLANs are not configured:
   Configure IGMP on the default VLAN (DEFAULT_VLAN; VID=1.)

2. If multiple VLANs are configured:
   Configure IGMP on a per-VLAN basis for every VLAN where the feature is of use.

Configuring and displaying IGMP (CLI)

show ip igmp vlan

Syntax

show ip igmp vlan vid

Description

Displays IGMP configuration for a specified VLAN or all VLANs on the switch.

Example output

switch(config)# show ip igmp vlan 1

IGMP Service Protocol Info

| Total VLANs with IGMP enabled | 30 |
| Current count of multicast groups joined | 20 |
| VLAN ID : 2 | |
| VLAN Name : VLAN2 | |
| IGMP version : 2 | |
| Querier Address : 10.255.128.2 | |
| Querier Port : A1 | |
Viewing the current IGMP configuration

Syntax
show ip igmp config

Description
Displays IGMP configuration for all VLANs on the switch.

Subcommands
show ip igmp vlan vid config
Displays IGMP configuration for a specific VLAN on the switch, including per-port data.

Example
Suppose you have the following VLAN and IGMP configurations on the switch:

<table>
<thead>
<tr>
<th>VLAN ID</th>
<th>VLAN name</th>
<th>IGMP enabled</th>
<th>Querier</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>DEFAULT_VLAN</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>22</td>
<td>VLAN-2</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>33</td>
<td>VLAN-3</td>
<td>No</td>
<td>Yes</td>
</tr>
</tbody>
</table>

You could use the CLI to display this data as follows:

Listing of IGMP configuration for all VLANs in the switch

switch(config)# show ip igmp config

IGMP Service Config

Control unknown multicast [Yes] : Yes
Forced fast leave timeout [0] : 4
Delayed flush timeout [0] : 0

<table>
<thead>
<tr>
<th>VLAN ID</th>
<th>VLAN Name</th>
<th>IGMP Enabled</th>
<th>Querier Allowed</th>
<th>Querier Interval</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>DEFAULT_VLAN</td>
<td>Yes</td>
<td>No</td>
<td>125</td>
</tr>
<tr>
<td>22</td>
<td>VLAN-2</td>
<td>Yes</td>
<td>Yes</td>
<td>125</td>
</tr>
<tr>
<td>33</td>
<td>VLAN-3</td>
<td>No</td>
<td>Yes</td>
<td>125</td>
</tr>
</tbody>
</table>

The following version of the show ip igmp command includes the VLAN ID (vid) designation, and combines the above data with the IGMP per-port configuration:

Listing of IGMP configuration for a specific VLAN
show ip igmp statistics

Use the command to view the IGMP high-level statistics for all VLANs on the switch.

Syntax

show ip igmp statistics

Displaying statistics for IGMP joined groups

```
switch(config)# show ip igmp statistics

IGMP Service Statistics

Total VLAN's with IGMP enabled: 33
Current count of multicast groups joined: 21

IGMP Joined Group Statistics

<table>
<thead>
<tr>
<th>VLAN ID</th>
<th>VLAN Name</th>
<th>Total</th>
<th>Filtered</th>
<th>Standard</th>
<th>Static</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>DEFAULT_VLAN</td>
<td>52</td>
<td>50</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>22</td>
<td>VLAN-2</td>
<td>80</td>
<td>75</td>
<td>5</td>
<td>0</td>
</tr>
<tr>
<td>33</td>
<td>VLAN-3</td>
<td>1100</td>
<td>1000</td>
<td>99</td>
<td>1</td>
</tr>
</tbody>
</table>
```

show ip igmp vlan vid

Use this command to view the IGMP historical counters for a VLAN.

Syntax:

show ip igmp vlan vid counters

Display of IGMP historical counters for a VLAN

```
switch(config)# show ip igmp vlan 1 counters

IGMP service Vlan counters

VLAN ID : 1
VLAN Name : DEFAULT_VLAN

  General Query Rx       : 58
  General Query Tx       : 58
```
show ip igmp groups

Use the command to view the IGMP group address information

Syntax:
show ip igmp groups

Displaying IGMP groups address information

```
switch(vlan-2)# show ip igmp groups
IGMP Group Address Information

<table>
<thead>
<tr>
<th>VLAN ID</th>
<th>Group Address</th>
<th>Expires</th>
<th>UpTime</th>
<th>Last Reporter</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>22</td>
<td>239.20.255.7</td>
<td>1h 2m 5s</td>
<td>1h 14m 5s</td>
<td>192.168.0.2</td>
<td>Filter</td>
</tr>
<tr>
<td>22</td>
<td>239.20.255.8</td>
<td>1h 2m 5s</td>
<td>1h 14m 5s</td>
<td>192.168.0.2</td>
<td>Standard</td>
</tr>
<tr>
<td>22</td>
<td>239.20.255.9</td>
<td>1h 2m 5s</td>
<td>1h 14m 5s</td>
<td>192.168.0.2</td>
<td>Static</td>
</tr>
</tbody>
</table>
```

show ip igmp vlan group

Syntax:
show ip igmp vlan vid group ip-addr

Example output

Here is an example of Group information for a VLAN with a filtered address group

```
switch(config)# show ip igmp vlan 22 group 239.20.255.7

IGMP Service Protocol Group Info

VLAN ID: 22
VLAN NAME: VLAN-2

Filtered Group Address: 239.20.255.7
Last Reporter: 192.168.0.2
Up Time: 1 hr 14 min 5 sec

<table>
<thead>
<tr>
<th>Port</th>
<th>Port Type</th>
<th>Port Mode</th>
<th>Expires</th>
<th>Access</th>
</tr>
</thead>
<tbody>
<tr>
<td>A1</td>
<td>100/1000T</td>
<td>Auto</td>
<td>1hr 2min 5 sec</td>
<td>Host</td>
</tr>
</tbody>
</table>
```

ip igmp

You can enable IGMP on a VLAN, along with the last-saved or default IGMP configuration (whichever was most recently set), or you can disable IGMP on a selected VLAN.

Syntax:
no ip igmp

Enables IGMP on a VLAN. This command must be executed in a VLAN context.

**Enable IGMP on VLAN 1**

```text
switch(vlan-1)# vlan 1 ip igmp
```

- or -

```text
ip igmp
```

**Disable IGMP on VLAN 1**

```text
switch(config)# no vlan 1 ip igmp
```

If you disable IGMP on a VLAN and then later re-enable IGMP on that VLAN, the switch restores the last-saved IGMP configuration for that VLAN. For more information, see the Management and Configuration Guide for your switch.

You can also combine the `ip igmp` command with other IGMP-related commands, as described in the following sections.

**vlan ip igmp**

**Syntax:**

```text
vlan vid ip igmp [auto port-list | blocked port-list | forward port-list]
```

Used in the VLAN context, specifies how each port should handle IGMP traffic.

Default: auto.

Where a static multicast filter is configured on a port, and an IGMP filter created by this command applies to the same port, the IGMP filter overrides the static multicast filter for any inbound multicast traffic carrying the same multicast address as is configured in the static filter. For more information, see the Management and Configuration Guide for your switch.

**Example:**

Suppose you want to configure IGMP as follows for VLAN 1 on the 100/1000T ports on a module in slot 1:

<table>
<thead>
<tr>
<th>Port Range</th>
<th>Setting</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>A1-A2</td>
<td>auto</td>
<td>Filter multicast traffic. Forward IGMP traffic to hosts on these ports that belong to the multicast group for which the traffic is intended. (Also forward any multicast traffic through any of these ports that is connected to a multicast router.)</td>
</tr>
<tr>
<td>A3-A4</td>
<td>forward</td>
<td>Forward all multicast traffic through this port.</td>
</tr>
<tr>
<td>A5-A6</td>
<td>blocked</td>
<td>Drop all multicast traffic received from devices on these ports.</td>
</tr>
</tbody>
</table>

For a description of the default behavior of data-driven switches, see Automatic fast-leave IGMP.
Depending on the privilege level, you could use one of the following commands to configure IGMP on VLAN 1 with the above settings:

```
switch(config)# vlan 1 ip igmp auto a1,a2 forward a3,a4
      blocked a5,a6
switch(vlan-1)# ip igmp auto a1,a2 forward a3,a4
      blocked a5,a6
```

The following command displays the VLAN and per-port configuration resulting from the above commands.

```
switch show ip igmp vlan 1 config
```

### vlan ip igmp querier

**Syntax:**

```
no vlan vid ip igmp querier
```

This command disables or re-enables the ability for the switch to become querier if necessary.

The `no` version of the command disables the querier function on the switch. The `show ip igmp config` command displays the current querier command.

Default querier capability: Enabled

### ip igmp querier interval

To specify the number of seconds between membership queries, enter this command with the desired interval.

**Syntax:**

```
no ip igmp querier interval [5-300]
```

This command must be issued in a VLAN context.

Specifies the number of seconds between membership queries. The `no` form of the command sets the interval to the default of 125 seconds.

Default: 125 seconds

For example, to set the querier interval to 300 seconds on ports in VLAN 8:

```
switch(vlan-8)# ip igmp querier interval 300
```

### ip igmp static-group

Use this command to configure a group on the switch so that multicast traffic for that group can be forwarded with a receiver host. Traffic will be flooded for this group.

**Syntax:**

```
no ip igmp static-group group-address
```

This command must be issued in a VLAN context.

Creates the IGMP static group with the specified group address on the selected VLAN. The `no` form of the command deletes the static group on the selected VLAN.

### ip igmp fastleave

For information, see Automatic fast-leave IGMP.
Syntax:
no ip igmp fastleave port-list
Enables IGMP fast-leaves on the specified ports in the selected VLAN.
The no form of the command disables IGMP fast-leave on the specified ports in the selected VLAN.
Use show running to display the ports per-VLAN on which fast-leave is disabled.
Default: Enabled

ip igmp forcedfastleave
For information about forced fast-leave, see Forced fast-leave IGMP.
Syntax:
no vlan vid ip igmp forcedfastleave port-list
Enables IGMP forced fast-leave on the specified ports in the selected VLAN, even if they are cascaded.
The no form of the command disables forced fast-leave on the specified ports in the selected VLAN.
Use show running to display the ports per-VLAN on which forced fast-leave is enabled.
Default: Disabled
show running-config
Displays a non-default IGMP forced fast-leave configuration on a VLAN. The show running-config output does not include forced fast-leave if it is set to the default of 0.
forcedfastleave
Can be used when there are multiple devices attached to a port.

igmp fastlearn
The fast learn option allows fast convergence of multicast traffic after a topology change. This command is executed in the global config context.
Syntax:
no igmp fastlearn port-list
This command enabled fast learn on the specified ports. The form of the command disables the fast learn function on the specified ports.
Default: Disabled

To enable fastlearn on ports 5 and 6

switch(config)# igmp fastlearn 5-6

show igmp delayed-flush
When enabled, this feature continues to filter IGMP groups for a specified additional time after IGMP leaves have been sent. The delay in flushing the group filter prevents unregistered traffic from being forwarded by the server during the delay period. In practice, this is rarely necessary on the switches, which support data-driven IGMP. (Data-driven IGMP, which is enabled by default, prunes off any unregistered IGMP streams detected on the switch.)

Syntax:
igmp delayed-flush time-period
Where leaves have been sent for IGMP groups, enables the switch to continue to flush the groups for a specified period of time. This command is applied globally to all IGMP-configured VLANs on the switch.
Range: 0 - 255; Default: Disabled (0)

**Syntax:**

```
show igmp delayed-flush
```

Displays the current igmp delayed-flush setting.

### Preventing unjoined multicast traffic

For more information about unjoined multicast traffic, see [Unjoined multicast traffic](#).

**Syntax:**

```
no igmp filter-unknown-mcast
```

Enables interface isolation for unjoined multicast groups. IGMP is configured so that each interface with IGMP enabled will have a data-driven multicast filter associated with it, preventing unjoined IP multicast packets from being flooded. A reboot is required for the change to take effect.
Default: Disabled

### Configuring IGMP proxy (CLI)

For more information on IGMP proxy, see [IGMP general operation and features](#).

### Adding or leaving a multicast domain

**Syntax:**

```
no igmp-proxy-domain domain-name [border-router-ip-address | mcast-range | all]
```

The no form of the command is used to remove a multicast domain.

All VLANs associated with the domain must first be removed for this command to work. See the `no form of igmp-proxy` in the VLAN context command.

- **domain-name**
  User-defined name to associate with the PIM border router and multicast range that is being sent toward the border router.

- **border-router-ip-addr**
  The IP address of the border router toward which IGMP proxy packets are sent. Not required for the no form of the command.

---

**NOTE**

The current routing FIB determines the best path toward the border router and therefore the VLAN that a proxy is sent out on

- **[low-bound-ip-address | all]**
  The low boundary (inclusive) of the multicast address range to associate with this domain (for example, 234.0.0.1.)

  If all is selected, the multicast addresses in the range of 224.0.1.0 to 239.255.255.255 are included in this domain.
Addresses 224.0.0.0 to 224.0.0.255 are never used, because these addresses are reserved for protocols.

high-bound-ip-address
The high boundary (inclusive) of the multicast address range to associate with this domain (for example, 236.1.1.1.)

**IGMP proxy border IP address command**
This example shows the IGMP proxy border IP addresses (111.11.111.111) being configured.

```
switch(config)# igmp-proxy-domain Bob 111.11.111.111
```

**Setting the lower and upper bounds for multicasting**
This example shows the lower and upper boundaries of the multicast address range associated with the domain named Bob.
```
switch(config)# igmp-proxy-domain Bob 111.11.111.111 234.0.0.1
switch(config)# igmp-proxy-domain Bob 111.11.111.111 236.1.1.1
```

**Informs the VLAN which IGMP proxy domains to use with joins on the VLAN**
This command is performed when in VLAN context mode. When a query occurs on the upstream interface, an IGMP join is sent for all multicast addresses that are currently joined on the downstream interface.

**Syntax:**
```
no igmp-proxy domain-name
```
The no version of the command with no domain name specified removes all domains associated with this VLAN.

Multiple different domains may be configured in the same VLAN context where the VLAN is considered the downstream interface. The domain name must already exist in the database before using this command for adding a domain.

**Note**
If the unicast routing path to the specified IP address was through the specified VLAN, no proxy IGMP would occur, that is, a proxy is not sent back out on the VLAN that the IGMP join came in on.

If no unicast route exists to the border router, no proxy IGMP packets are sent.

**Viewing the IGMP proxy data**

**Syntax:**
```
show igmp-proxy [entries | domains | vlans]
```
Shows the currently active IGMP proxy entries, domains, or VLANs.

**Showing active IGMP proxy entries**
```
switch(config)# show igmp-proxy entries
Total number of multicast routes: 2
Multicast Address Border Address VID Multicast Domain
```

**Syntax:**
```
no igmp-proxy domain-name
```
The no version of the command with no domain name specified removes all domains associated with this VLAN.
Showing IGMP proxy domains

switch(config)# show igmp-proxy domains

Total number of multicast domains: 5

<table>
<thead>
<tr>
<th>Multicast Domain</th>
<th>Multicast Range</th>
<th>Border Address</th>
<th>Active entries</th>
</tr>
</thead>
<tbody>
<tr>
<td>George</td>
<td>225.1.1.1/234.43.209.12</td>
<td>192.168.1.1</td>
<td>2</td>
</tr>
<tr>
<td>SAM</td>
<td>235.0.0.0/239.1.1.1</td>
<td>15.43.209.1</td>
<td>1</td>
</tr>
<tr>
<td>Jane</td>
<td>236.234.1.1/236.235.1.1</td>
<td>192.160.1.2</td>
<td>0</td>
</tr>
<tr>
<td>Bill</td>
<td>ALL</td>
<td>15.43.209.1</td>
<td>0</td>
</tr>
</tbody>
</table>

Showing active IGMP proxy VLANs

switch(config)# show igmp-proxy vlans

IGMP PROXY VLANS

<table>
<thead>
<tr>
<th>VID</th>
<th>Multicast Domain</th>
<th>Active entries</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>George</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>Sam</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>Jane</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>George</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>George</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>Bill</td>
<td>0</td>
</tr>
</tbody>
</table>

IGMP general operation and features

In a network where IP multicast traffic is transmitted for various multimedia applications, you can use the switch to reduce unnecessary bandwidth usage on a per-port basis by configuring IGMP. In the factory default state (IGMP disabled), the switch simply floods all IP multicast traffic it receives on a given VLAN through all ports on that VLAN (except the port on which it received the traffic.) This can result in significant and unnecessary bandwidth usage in networks where IP multicast traffic is a factor. Enabling IGMP allows the ports to detect IGMP queries and report packets and manage IP multicast traffic through the switch.

IGMP is useful in multimedia applications such as LAN TV, desktop conferencing, and collaborative computing, where there is multipoint communication, that is, communication from one to many hosts, or communication originating from many hosts and destined for many other hosts. In such multipoint applications, IGMP is configured on the hosts, and multicast traffic is generated by one or more servers (inside or outside of the local network.) Switches in the network (that support IGMP) can then be configured to direct the multicast traffic to only the ports where needed. If multiple VLANs are configured, you can configure IGMP on a per-VLAN basis. Enabling IGMP allows detection of IGMP queries and report packets used to manage IP multicast traffic through the switch. If no other querier is detected, the switch then also functions as the querier. If you need to disable the querier feature, you can do so using the IGMP configuration CLI commands, see `vlan ip igmp querier`. 

---

234.43.209.12 192.168.1.1 1 George
235.22.22.12 15.43.209.1 1 SAM
226.44.3.3 192.168.1.1 2 George
IGMP configuration on the switches operates at the VLAN context level. If you are not using VLANs, configure IGMP in VLAN 1 (the default VLAN) context.

More information
Overview of IGMP on page 10

Options
With the CLI, you can configure these additional options:

Forward with high priority
Disabling this parameter (the default) causes the switch or VLAN to process IP multicast traffic, along with other traffic, in the order received (usually, normal priority.) Enabling this parameter causes the switch or VLAN to give a higher priority to IP multicast traffic than to other traffic.

Auto/blocked/forward
You can use the console to configure individual ports to any of the following states:

Auto
(Default) Causes the switch to interpret IGMP packets and to filter IP multicast traffic based on the IGMP packet information for ports belonging to a multicast group. This means that IGMP traffic will be forwarded on a specific port only if an IGMP host or multicast router is connected to the port.

Blocked
Causes the switch to drop all IGMP transmissions received from a specific port, and also blocks all outgoing IP Multicast packets for that port, thus preventing IGMP traffic from moving through specific ports.

Forward
Causes the switch to forward all IGMP and multicast transmissions through the port.

Operation with or without IP addressing
This feature helps to conserve IP addresses by enabling IGMP to run on VLANs that do not have an IP address. See Operation with or without IP addressing.

Querier capability
The switch performs querier function for IGMP on VLANs having an IP address when there is no other device in the VLAN acting as querier. See About using the switch as querier.

Whenever IGMP is enabled, the switch generates an Event Log message only after the querier election.

IP multicast traffic groups are identified by IP addresses in the range of 224.0.0.0 to 239.255.255.255. Also, incoming IGMP packets intended for reserved, or "well-known" multicast addresses, automatically flood through all ports (except the port on which the packets entered the switch.)

Number of IP multicast addresses allowed
The number of IGMP filters (addresses) and static multicast filters available is 2,038. Additionally, 16 static multicast filters are allowed, If multiple VLANs are configured, then each filter is counted once per VLAN in which it is used.

The following table shows the number of IGMP and static multicast filters available per platform.
Table 2: *IP multicast address per platform*

<table>
<thead>
<tr>
<th>Multicast Group Filters</th>
<th>5400R</th>
<th>3810M</th>
<th>3800</th>
</tr>
</thead>
<tbody>
<tr>
<td>IPv4</td>
<td>2038</td>
<td>2038</td>
<td>2038</td>
</tr>
<tr>
<td>IPv6</td>
<td>2037</td>
<td>2037</td>
<td>2037</td>
</tr>
</tbody>
</table>

**How IGMP operates**

IGMP is an internal protocol of the IP suite. IP manages multicast traffic by using switches, multicast routers, and hosts that support IGMP. A multicast router is not necessary as long as a switch is configured to support IGMP with the querier feature enabled. A set of hosts, routers, and/or switches that send or receive multicast data streams to or from the same sources is called a **multicast group**, and all devices in the group use the same multicast group address. The multicast group running version 2 of IGMP uses three fundamental types of messages to communicate:

**Query**

A message sent from the querier (multicast router or switch) asking for a response from each host belonging to the multicast group. If a multicast router supporting IGMP is not present, the switch must assume this function to elicit group membership information from the hosts on the network. If you need to disable the querier feature, do so through the CLI using the IGMP configuration CLI commands.

**Report (Join)**

A message sent by a host to the querier to indicate that the host wants to be or is a member of a given group indicated in the report message.

**Leave group**

A message sent by a host to the querier to indicate that the host has ceased to be a member of a specific multicast group.

---

**Note on IGMP version 3 support**

When an IGMPv3 Join is received by the switch, it accepts the host request and begins to forward the IGMP traffic. This means that ports that have not joined the group and are not connected to routers or the IGMP Querier will not receive the group's multicast traffic.

The switch does not support the IGMPv3 "Exclude Source" or "Include Source" options in the Join Reports. Rather, the group is simply joined from all sources.

An IP multicast packet includes the multicast group (address) to which the packet belongs. When an IGMP client connected to a switch port needs to receive multicast traffic from a specific group, it joins the group by sending an IGMP report (join request) to the network. (The multicast group specified in the join request is determined by the requesting application running on the IGMP client.) When a networking device with IGMP enabled receives the join request for a specific group, it forwards any IP multicast traffic it receives for that group through the port on which the join request was received. When the client is ready to leave the multicast group, it sends a Leave
Group message to the network and ceases to be a group member. When the leave request is detected, the appropriate IGMP device ceases transmitting traffic for the designated multicast group through the port on which the leave request was received (as long as there are no other current members of that group on the affected port.) Thus, IGMP identifies members of a multicast group (within a subnet) and allows IGMP-configured hosts (and routers) to join or leave multicast groups.

To display IGMP data showing active group addresses, reports, queries, querier access port, and active group address data (port, type, and access), see the Management and Configuration Guide for your switch.

**Operation with or without IP addressing**

You can configure IGMP on VLANs that do not have IP addressing. The benefit of IGMP without IP addressing is a reduction in the number of IP addresses you have to use and configure. This can be significant in a network with a large number of VLANs. The limitation on IGMP without IP addressing is that the switch cannot become Querier on any VLANs for which it has no IP address—so the network administrator must ensure that another IGMP device will act as Querier. It is also advisable to have an additional IGMP device available as a backup Querier.

<table>
<thead>
<tr>
<th>IGMP function available with IP addressing configured on the VLAN</th>
<th>Available without IP addressing?</th>
<th>Operating differences without an IP address</th>
</tr>
</thead>
<tbody>
<tr>
<td>Forward multicast group traffic to any port on the VLAN that has received a join request for that multicast group.</td>
<td>Yes</td>
<td>None</td>
</tr>
<tr>
<td>Forward join requests (reports) to the Querier.</td>
<td>Yes</td>
<td>None</td>
</tr>
<tr>
<td>Configure individual ports in the VLAN to Auto (the default) Blocked, or Forward.</td>
<td>Yes</td>
<td>None</td>
</tr>
<tr>
<td>Configure IGMP traffic forwarding to normal or high-priority forwarding.</td>
<td>Yes</td>
<td>None</td>
</tr>
<tr>
<td>Age-out IGMP group addresses when the last IGMP client on a port in the VLAN leaves the group.</td>
<td>Yes</td>
<td>Requires that another IGMP device in the VLAN has an IP address and can operate as Querier. This can be a multicast router or another switch configured for IGMP operation. (Hewlett Packard Enterprise recommends that the VLAN also include a device operating as a backup Querier in case the device operating as the primary Querier fails for any reason.)</td>
</tr>
<tr>
<td>Support Fast-Leave IGMP and Forced Fast-Leave IGMP (below.)</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>Support automatic Querier election.</td>
<td>No</td>
<td>Querier operation not available.</td>
</tr>
<tr>
<td>Operate as the Querier.</td>
<td>No</td>
<td>Querier operation not available.</td>
</tr>
<tr>
<td>Available as a backup Querier.</td>
<td>No</td>
<td>Querier operation not available.</td>
</tr>
</tbody>
</table>
Automatic fast-leave IGMP

Depending on the switch model, fast-leave is enabled or disabled in the default configuration. On switches that do not support data-driven IGMP, unregistered multicast groups are flooded to the VLAN rather than pruned. In this scenario, fast-leave IGMP can increase the problem of multicast flooding by removing the IGMP group filter before the Querier has recognized the IGMP leave. The Querier will continue to transmit the multicast group during this short time, and because the group is no longer registered, the switch will then flood the multicast group to all ports.

On switches that do support data-driven IGMP ("Smart" IGMP), when unregistered multicasts are received the switch automatically filters (drops) them. Thus, the sooner the IGMP leave is processed, the sooner this multicast traffic stops flowing.

Because of the multicast flooding problem mentioned above, the IGMP fast-leave feature is disabled by default on all switches that do not support data-driven IGMP (see the table above.) The feature can be enabled on these switches via an SNMP set of this object:

```
hpSwitchIgmpPortForceLeaveState.vid . port number
```

However, Hewlett Packard Enterprise does not recommend this, because it will increase the amount of multicast flooding during the period between the client’s IGMP leave and the Querier’s processing of that leave. For more information on this topic, see Forced fast-leave IGMP.

If a switch port has the following characteristics, the fast-leave operation will apply:

- Connected to only one end node.
- The end node currently belongs to a multicast group, that is, is an IGMP client.
- The end node subsequently leaves the multicast group.

Then the switch does not need to wait for the Querier status update interval, but instead immediately removes the IGMP client from its IGMP table and ceases transmitting IGMP traffic to the client. (If the switch detects multiple end nodes on the port, automatic fast-leave does not activate—regardless of whether one or more of these end nodes are IGMP clients.)

In the following figure, automatic fast-leave operates on the switch ports for IGMP clients "3A" and "5A," but not on the switch port for IGMP clients "7A" and "7B," server "7C," and printer "7D."

**Example of automatic fast-leave IGMP criteria**

When client "3A" running IGMP is ready to leave the multicast group, it transmits a Leave Group message. Because the switch knows that there is only one end node on port A3, it removes the client from its IGMP table and halts multicast traffic (for that group) to port A3. If the switch is not the Querier, it does not wait for the actual Querier to verify that there are no other group members on port A3. If the switch itself is the Querier, it does not query port A3 for the presence of other group members.

Fast-leave operation does not distinguish between end nodes on the same port that belong to different VLANs. Thus, for example, even if all of the devices on port A6 in the figure belong to different VLANs, fast-leave does not operate on port A6.
Default (enabled) IGMP operation solves the "delayed leave" problem
Fast-leave IGMP is enabled by default. When fast-leave is disabled and multiple IGMP clients are connected to the same port on an IGMP device (switch or router), if only one IGMP client joins a given multicast group, then later sends a Leave Group message and ceases to belong to that group, the switch automatically retains that IGMP client in its IGMP table and continues forwarding IGMP traffic to the IGMP client until the Querier triggers confirmation that no other group members exist on the same port. This delayed leave operation means that the switch continues to transmit unnecessary multicast traffic through the port until the Querier renews multicast group status.

Forced fast-leave IGMP
When enabled, forced fast-leave IGMP speeds up the process of blocking unnecessary IGMP traffic to a switch port that is connected to multiple end nodes. (This feature does not activate on ports where the switch detects only one end node.) For example, in Automatic fast-leave IGMP, even if you configured forced fast-leave on all ports in the switch, the feature would activate only on port A6 (which has multiple end nodes) when a Leave Group request arrived on that port.

When a port having multiple end nodes receives a Leave Group request from one end node for a given multicast group "X," forced fast-leave activates and waits a small amount of time to receive a join request from any other group "X" member on that port. If the port does not receive a join request for that group within the forced-leave interval, the switch then blocks any further group "X" traffic to the port.

Fast learn
The fast learn option allows fast convergence of multicast traffic after a topology change.

To enable fastlearn on ports 5 and 6

```
switch(config)# igmp fastlearn 5-6
```

Unjoined multicast traffic
This feature adds a global IGMP multicast configuration option to the switch that results in each VLAN having a multicast filter. The filter prevents unjoined multicast traffic from being forwarded on interfaces associated with IGMP queriers. Each filter only contains interfaces that are queriers on the same VLAN, so multicast traffic is only flooded on interfaces that contain queriers that are on the same VLAN as the multicast traffic.

On switch bootup, all VLANs that are IGMP-enabled are guaranteed one multicast filter. You can always reboot the switch to recreate this configuration where each IGMP-enabled VLAN has a multicast filter.

```
NOTE
Joined multicast traffic continues to be forwarded as usual.
```

You must reboot the switch after configuring the per-VLAN filter.

Enabling the IGMP multicast filter

```
switch(config)# igmp filter-unknown-mcast
Command will take effect after saving configuration and reboot.
```

The following example shows the multicast traffic being flooded to all queriers on all VLANs; this is the default behavior. The `igmp filter-unknown-mcast` command has not been executed.
**Table 4: Multicast filter table on distribution switch**

<table>
<thead>
<tr>
<th>VLAN ID</th>
<th>Member Ports</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 (all VLANs)</td>
<td>1, 2, 3</td>
</tr>
</tbody>
</table>

**Figure 1 Example of unknown multicast traffic flooding on all ports connected to a querier for any VLAN**

In the following example, igmp filter-unknown-mcast has been configured. The multicast traffic only goes to the querier on the same VLAN as the multicast server.

**Table 5: Multicast filter table on distribution switch**

<table>
<thead>
<tr>
<th>VLAN ID</th>
<th>Member Ports</th>
</tr>
</thead>
<tbody>
<tr>
<td>100</td>
<td>1</td>
</tr>
<tr>
<td>200</td>
<td>2</td>
</tr>
<tr>
<td>300</td>
<td>3</td>
</tr>
</tbody>
</table>
Figure 2  Example of unknown multicast traffic not flooding out ports connected to queriers in separate VLANs

To display the status of IGMP multicast filtering use the show ip igmp command. If the IGMP Filter Unknown Multicast setting is different from the IGMP Filter Unknown Multicast status, a reboot is required to activate the desired setting. This setting will then be reflected in the status.

**IGMP unknown multicast filter setting being enabled but not yet activated**

```
switch(config)# show igmp filter-unknown-mcast
IGMP Filter Unknown Multicast: Enabled
IGMP Filter Unknown Multicast Status: Disabled
```

To display information about IGMP multicast filtering by interface, use the `show ip igmp` command.

**IGMP proxy forwarding**

When a network has a border router connecting a PIM-SM domain to a PIM-DM domain, the routers that are completely within the PIM-DM domain have no way to discover multicast flows in the PIM-SM domain. When an IGMP join occurs on a router entirely within the PIM-DM domain for a flow that originates within the PIM-SM domain, it is never forwarded to the PIM-SM domain.

The IGMP proxy is a way to propagate IGMP joins across router boundaries. The proxy triggers the boundary router connected to a PIM-SM domain to query for multicast flows and forward them to the PIM-DM domain. IGMP needs to be configured on all VLAN interfaces on which the proxy is to be forwarded or received, and PIM-DM must be running for the traffic to be forwarded.

You can configure an IGMP proxy on a selected VLAN that will forward IP joins (reports) and IGMP leaves to the upstream border router between the two multicast domains. You must specify the VLANs on which the proxy is enabled as well as the address of the border router to which the joins are forwarded.

**How IGMP proxy forwarding works**

The following steps illustrate how to flood a flow from the PIM-SM domain into the PIM-DM domain when an IGMP join for that flow occurs in the PIM-DM domain. See figure.
**Procedure**

1. Configure Routing Switch 1 with the IGMP proxy forwarding function to forward joins toward Border Router 1; in addition, configure Routing Switch 1 to forward joins from VLAN 1 toward Border Router 2, as is VLAN 4 on Routing Switch 3.

2. Configure VLAN 2 on Routing Switch 2 to forward joins toward Border Router 1.

3. When the host connected in VLAN 1 issues an IGMP join for multicast address 235.1.1.1, the join is proxied by Routing Switch 1 onto VLAN 2 and VLAN 4. The routing information table in Routing Switch 1 indicates that the packet to Border Router 1 and Border Router 2 is on VLAN 2 and VLAN 4, respectively. *IGMP proxy example*

4. Routing Switch 2 then proxies the IGMP join into VLAN 3, which is connected to Border Router 1.

5. Border Router 1 uses PIM-SM to find and connect to the multicast traffic for the requested traffic. The traffic is flooded into the PIM-DM network where it is routed to the original joining host.

6. Additionally, the join was proxied from Routing Switch 3 to Border Router 2. At first, both border routers will flood the traffic into the PIM-DM domain. However, PIM-DM only forwards multicasts based on the shortest reverse path back to the source of the traffic as determined by the unicast routing tables (routing FIB.) Only one multicast stream is sent to the joining host. This configuration provides a redundant in case the first fails.

**Operating notes for IGMP proxy forwarding**

- You can configure up to 12 multicast domains, which indicate a range of multicast addresses and the IP address of the PIM-SM/PIM-DM border router.
- You must give each domain a unique name, up to 20 characters.
- The domains may have overlapping multicast ranges.
The IP address of the border router may be the same or different in each configured domain.
- Duplicate IGMP joins are automatically prevented, or leaves that would remove a flow currently joined by multiple hosts.
- Range overlap allows for redundant connectivity and the ability for multicasts to arrive from different border routers based on the shortest path back to the source of the traffic.
- The configured domain names must be associated with one or more VLANs for which the proxy joins are to be done.
- All routers in the path between the edge router receiving the initial IGMP packets and the border router have to be configured to forward IGMP using IGMP proxy.
- All upstream and downstream interfaces using IGMP proxy forwarding require IGMP and PIM to be enabled.
- You must remove all VLAN associations with the domain name before that domain name can be removed.
- The appropriate border routers must be used for each VLAN, or PIM-DM will not forward the traffic. This could occur when multiple border routers exist. It may be necessary to configure multiple overlapping domains if the multicast source address can generate the same multicast address and have different best paths to the PIM-DM domain.

Be careful to avoid configuring a IGMP forward loop, because this would leave the VLANs in a joined state forever once an initial join is sent from a host. For example, a join is issued from the host in VLAN 2 and Routing Switch 2 will proxy the join onto VLAN 1. Routing Switch 3 will then proxy the join back onto VLAN 2 and increment its internal count of the number of joins on VLAN 2. Even after the host on VLAN 2 issues a leave, the proxy join will continue to remain and refresh itself each time a query occurs on VLAN 2. This type of loop could be created with multiple routers if an IGMP proxy is allowed to get back to the VLAN of the router that initially received the IGMP join from a host as shown in the following figure.

Proxy loop scenario

About using the switch as querier
The function of the IGMP Querier is to poll other IGMP-enabled devices in an IGMP-enabled VLAN to elicit group membership information. The switch performs this function if there is no other device in the VLAN, such as a multicasting router, to act as Querier. Although the switch automatically ceases Querier operation in an IGMP-enabled VLAN if it detects another Querier on the VLAN, you can also use the switch’s CLI to disable the Querier capability for that VLAN.

A Querier is required for proper IGMP operation. For this reason, if you disable the Querier function on a switch, ensure that there is an IGMP Querier (and, preferably, a backup Querier) available on the same VLAN.

If the switch becomes the Querier for a particular VLAN (for example, the DEFAULT_VLAN), then subsequently detects queries transmitted from another device on the same VLAN, the switch ceases to operate as the Querier for that VLAN. If this occurs, the switch Event Log lists a pair of messages similar to these:

```
I 01/15/12 09:01:13 igmp: DEFAULT_VLAN: Other Querier detected
I 01/15/12 09:01:13 igmp: DEFAULT_VLAN: This switch is no longer Querier
```

In the above scenario, if the other device ceases to operate as a Querier on the default VLAN, the switch detects this change and can become the Querier as long as it is not pre-empted by some other IGMP Querier on the VLAN. In this case, the switch Event Log lists messages similar to the following to indicate that the switch has become the Querier on the VLAN:

```
I 01/15/12 09:21:55 igmp: DEFAULT_VLAN: Querier Election in process
I 01/15/12 09:22:00 igmp: DEFAULT_VLAN: This switch has been elected as Querier
```

**Well-known or reserved multicast addresses excluded from IP multicast filtering**

Each multicast host group is identified by a single IP address in the range of 224.0.0.0 through 239.255.255.255. Specific groups of consecutive addresses in this range are termed “well-known” addresses and are reserved for predefined host groups. IGMP does not filter these addresses, so any packets the switch receives for such addresses are flooded out all ports assigned to the VLAN on which they were received (except the port on which the packets entered the VLAN.)

The following table lists the 32 well-known address groups (8192 total addresses) that IGMP does not filter on.

**Table 6: IP multicast address groups excluded from IGMP filtering**

<table>
<thead>
<tr>
<th>Groups of consecutive addresses in the range of 224.0.0.X to 239.0.0.X</th>
<th>Groups of consecutive addresses in the range of 224.128.0.X to 239.128.0.X</th>
</tr>
</thead>
<tbody>
<tr>
<td>224.0.0.x</td>
<td>232.0.0.x</td>
</tr>
<tr>
<td>225.0.0.x</td>
<td>233.0.0.x</td>
</tr>
<tr>
<td>226.0.0.x</td>
<td>234.0.0.x</td>
</tr>
<tr>
<td>227.0.0.x</td>
<td>235.0.0.x</td>
</tr>
<tr>
<td>228.0.0.x</td>
<td>236.0.0.x</td>
</tr>
<tr>
<td>229.0.0.x</td>
<td>237.0.0.x</td>
</tr>
</tbody>
</table>
With aliasing limitation associated with MAC mode, certain non reserved multicast IP addresses are displayed as "reserved" addresses.

For example: 225.0.0.x Multicast IP address is aliased to 224.0.0.x to be displayed as "reserved".

**IP multicast filters**

IP multicast addresses occur in the range from 224.0.0.0 through 239.255.255.255 (which corresponds to the ethernet multicast address range of 01005e-000000 through 01005e-7ffff.) Where a switch has a static traffic/security filter configured with a "multicast" filter type and a "multicast address" in this range, the switch will use the static filter unless IGMP learns of a multicast group destination in this range. In this case, IGMP dynamically takes over the filtering function for the multicast destination addresses for as long as the IGMP group is active. If the IGMP group subsequently deactivates, the switch returns filtering control to the static filter.

**Reserved addresses excluded from IP multicast (IGMP) filtering**

Traffic to IP multicast groups in the IP address range of 224.0.0.0 to 224.0.0.255 will always be flooded because addresses in this range are "well known" or "reserved" addresses. Thus, if IP multicast is enabled, and there is an IP multicast group within the reserved address range, traffic to that group will be flooded instead of filtered by the switch.

**Multicast ARP support**

To support IP multicasting, the multicast range of 01-00-5E-00-00-00 to 01-00-5E-7F-FF-FF is reserved for Ethernet MAC addresses. The command `ip arp-mcast-replies` enables acceptance of the MAC addresses in the IP multicast range.

**Syntax:**
no ip arp-mcast-replies

Enables acceptance of multicast MAC addresses in the IP multicast address range in ARP requests and replies.

Default: Disabled

**Example:**
Switch(config)# ip arp-mcast-replies

**IGMPv3**

The Internet Group Management Protocol (IGMP) is used by IPv4 systems (hosts and routers) to report their IP multicast group membership to any neighboring multicast routers. This chapter is to describe version 3 of IGMP.
Version 1, specified in [RFC-1112], was the first widely-deployed version. Version 2, specified in [RFC-2236], added support for "low leave latency", that is, a reduction in the time it takes for a multicast router to learn that there are no longer any members of a particular group present on an attached network. Version 3 adds support for "source filtering", that is, the ability for a system to report interest in receiving packets "only" from specified source addresses, or from "all but" specified source addresses, sent to a particular multicast address. Version 3 is designed to be interoperable with Versions 1 and 2. If IGMPv3 report is sent to IGMPv2 router acting as querier, it will process the report as IGMPv2 report, you can ignore the source-specific details.

**Basic topology and configuration for IGMPv3**

![Basic topology and configuration for IGMPv3](image)

### Table 7: IGMPv3 configuration for Basic topology and configuration for IGMPv3

<table>
<thead>
<tr>
<th>DUT-1 configurations</th>
<th>DUT-2 configurations</th>
</tr>
</thead>
<tbody>
<tr>
<td>DUT-1(config)#igmp lookup-mode ip</td>
<td>DUT-2(config)#igmp lookup-mode ip</td>
</tr>
<tr>
<td>DUT-1(config)#vlan 60 ip address 60.0.0.1/24</td>
<td>DUT-2(config)#vlan 60 ip address 60.0.0.2/24</td>
</tr>
<tr>
<td>DUT-1(config)#vlan 60 ip igmp version 3</td>
<td>DUT-2(config)#vlan 60 ip igmp version 3</td>
</tr>
<tr>
<td>DUT-2(config)#no vlan 60 ip igmp querier</td>
<td></td>
</tr>
</tbody>
</table>

In the preceding figure, DUT-1 becomes the igmpv3 querier. Client-1 start receiving multicast traffic for group 235.6.6.6 from source 60.0.0.100 and client-2 start receiving multicast traffic for group 235.6.6.6 from source 60.0.0.200.

**NOTE**

If multiple igmp version devices are available in the network, the igmp querier device must have the lower version of IGMP. This can be achieved by executing the `no ip igmp querier` command under the `vlan` context on other devices.

### IGMPv3 commands

---

Aruba 3810M/5800R | Multicasting and Routing Guide
**igmp lookup-mode**

To first configure IGMPv3, the `igmp lookup-mode` must be changed from the default mac mode to ip mode. Use the `ip igmp lookup-mode` command to set the IGMP snooping lookup mode.

- **NOTE**
  IGMPv2 works both in ip mode and mac mode. Lookup-mode is applicable with IGMP disabled on all VLANs.

**Syntax**

```
ip igmp lookup-mode
```

**Options**

`mac`: Uses MAC look-up. (Default value)

`ip`: Uses IP look-up.

**igmp reload**

This command is used to reset the IGMP state on all interfaces.

**Syntax**

```
igmp reload
```

**Example output**

IGMP application is in Error State as System Resources are exhausted. Traffic will flood. Please disable IGMP on all VLANs or Issue the Command "igmp reload" to take it out of Error. Refer to your product manual for information on IGMP resource consumption.

This is the output for `igmp reload`.

**ip igmp**

You can enable IGMP on a VLAN, along with the last-saved or default IGMP configuration (whichever was most recently set), or you can disable IGMP on a selected VLAN.

**Syntax:**

```
no ip igmp
```

Enables IGMP on a VLAN. This command must be executed in a VLAN context.

**Enable IGMP on VLAN 1**

```
switch(vlan-1)# vlan 1 ip igmp
```

- or -

```
ip igmp
```

**Disable IGMP on VLAN 1**

```
switch(config)# no vlan 1 ip igmp
```

If you disable IGMP on a VLAN and then later re-enable IGMP on that VLAN, the switch restores the last-saved IGMP configuration for that VLAN. For more information, see the *Management and Configuration Guide* for your switch.
You can also combine the `ip igmp` command with other IGMP-related commands, as described in the following sections.

**ip igmp version**

This command sets the IGMP version and completes `igmpv3` configuration, enabling `igmpv3` on the switch. Note that the default value is 2.

**Syntax**

```
ip igmp version
no ip igmp version
```

**Parameters**

```
<2-3>: The protocol version to use; the default is 2.
no: resets the version to 2.
```

**ip igmp last-member-query-interval**

**Syntax**

```
ip igmp last-member-query-interval
no ip igmp last-member-query-interval
```

**Parameters**

```
<1-2>: The number of seconds between successive group-specific query messages; the default is 1.
```

The `no` version resets the value to its default value of 1 second.

**ip igmp querier**

By default, IGMP querier is enabled. To disable querier functionality, use the following command:

**Syntax**

```
ip igmp querier
```

**Parameters**

```
interval: Sets the interval in seconds between IGMP queries; the default is 125.
```

**ip igmp query-max-response-time**

**Syntax**

```
ip igmp query-max-response-time
no ip igmp query-max-response-time
```

**Parameters**

```
<10-128>: The number of seconds to wait for a response to a query; the default value is 10.
```

The `no` version resets the value to its default value of 10 seconds.

---

*NOTE: When using IGMPv2, the maximum query-max-response-time is 25.5 seconds. This is because IGMPv2 uses an 8-bit field and is expressed in units of tenths of a second.*

**ip igmp robustness**
Syntax
ip igmp robustness
no ip igmp robustness

Parameters
<1-8>: The number of times to retry a query; the default is 2.
The no version resets the value to its default value of 2.

show ip igmp
This command is used to show IGMP information for all VLANs

Syntax
show ip igmp

Example output
Below is the output when version is set to 3.

```
switch(vlan-60)# show ip igmp

IGMP Service Protocol Info

Total VLANs with IGMP enabled : 1
Current count of multicast groups joined : 2

IGMP Filter Unknown Multicast: Disabled
IGMP Filter Unknown Multicast Status: Disabled

VLAN ID : 1
VLAN Name : DEFAULT_VLAN
IGMP version : 2
IGMP is not enabled

VLAN ID : 60
VLAN Name : VLAN60
IGMP version : 3
Querier Address : 60.0.0.1
Querier Port : 23
Querier UpTime : 0h 10m 9s
Querier Expiration Time : 0h 3m 34s

Active Group Addresses Tracking Vers Mode Uptime Expires
----------------------- ------ ---- ------ ----- -------
235.6.6.6 Filter 3 INC 0m 3s 4m 17s
235.6.6.7 Filter 3 EXC 0m 3s 4m 16s

Sample configuration is as shown:

switch(vlan-60)# show run

Running configuration:

; JL253A Configuration Editor; Created on release #WC.16.02.0000x
; Ver #0d13.ef.7c.5f.fc.6b.fb.9f.fc.f3.ff.37.ef:09

hostname "switch-name"
module 1 type jl253a
igmp lookup-mode ip
snmp-server community "public" unrestricted
vlan 1
show ip igmp vlan 1
This command is used to show IGMP information for a VLAN.

Syntax
show ip igmp vlan 1

Example output
Below is the output when version is set to 3.

switch(vlan-60)# show ip igmp vlan 60

IGMP Service Protocol Info

Total VLANs with IGMP enabled : 1
Current count of multicast groups joined : 2

IGMP Filter Unknown Multicast: Disabled
IGMP Filter Unknown Multicast Status: Disabled

VLAN ID : 60
VLAN Name : VLAN60
IGMP version : 3
Querier Address : 60.0.0.1
Querier Port : 23
Querier UpTime : 0h 11m 44s
Querier Expiration Time : 0h 4m 5s

| Active Group Addresses Tracking Vers Mode Uptime Expires |
|----------------|---------|-------|--------|----------|
| 235.6.6.6       | Filter  | 3     | INC    | 1m 38s   | 4m 13s   |
| 235.6.6.7       | Filter  | 3     | EXC    | 1m 38s   | 4m 19s   |

show ip igmp vlan group

Syntax:
show ip igmp vlan vid group ip-addr

Example output
Here is an example of Group information for a VLAN with a filtered address group

switch(config)# show ip igmp vlan 22 group 239.20.255.7

IGMP Service Protocol Group Info

VLAN ID: 22
VLAN NAME: VLAN-2
show ip igmp vlan group source

This command is used to show IGMP group/source information for a VLAN.

Syntax

show ip igmp vlan <vid> group <ip4-addr> source <ip4-addr>

Example output

Below is the output when version is set to 3.

```
switch(vlan-60)# show ip igmp vlan 60 group 235.6.6.6 source 60.0.0.100
VLAN ID : 60      VLAN Name : VLAN60
Group Address : 235.6.6.6
Source Address : 60.0.0.100
Source Type : Filter

Port Mode Uptime   Expires   Configured Mode
----- ---- -------   ---------   ------------
    1   INC   3m 31s  2m 20s   auto
```

Usage errors

<table>
<thead>
<tr>
<th>Error condition</th>
<th>Error message</th>
</tr>
</thead>
<tbody>
<tr>
<td>Attempt to pass a nonexistent group</td>
<td>ipv4 address Group address is not found.</td>
</tr>
</tbody>
</table>

show ip igmp vlan group port

This command is used to show IGMP group/source information for a VLAN port.

Syntax

show ip igmp vlan <vid> group <ip4-addr> port <port>

Example output

Below is the output when version is set to 3.

```
switch(vlan-60)# show ip igmp vlan 60 group 1
VLAN ID : 60      Name : VLAN60
Group Address : 235.6.6.6
Last Reporter : 10.255.128.1
Group Type : Filter

Port Vers Mode Uptime   Expires   Timer   Timer   Timer   Forwarded Blocked
```
**show ip igmp vlan counters**

This command is used to show IGMP counters for a VLAN.

**Syntax**

`show ip igmp vlan <vid> counters`

**Example output**

Below is the output when version is set to 3.

```
switch(vlan-60)# show ip igmp vlan 60 counters

IGMP service VLAN counters

VLAN ID : 60  NAME : VLAN60

            Rx          Tx
            --------      --------
V1  All Hosts Query  0              0
V2  All Hosts Query  0              0
V3  All Hosts Query  12             0
V1  Group Specific Query  0          0
V2  Group Specific Query  0          0
V3  Group Specific Query  8           0
Group and Source Specific Query  12    0
V3  Member Report    22            22
V2  Member Report    8             0
V1  Member Report    0             0
V2  Member Leave     0             0
Forward to Routers   0            32
Forward to VLAN     0            26

Errors:

Unknown IGMP Type  0
Unknown Packet     0
Malformed Packet   0
Bad Checksum       0
Martian Source     0
Packet received on IGMP-disabled Interface 0
Interface Wrong Version Query 0

Port Counters:

Fast Leave : 4
Forced Fast Leave : 0
Membership Timeout : 0
```

**show ip igmp vlan statistics**

This command is used to show IGMP statistics for a VLAN.
Syntax

show ip igmp vlan <vid> statistics

Example output

Below is the output when version is set to 3.

```
switch(vlan-60)# show ip igmp vlan 60 statistics

IGMP Statistics

VLAN ID : 60
VLAN Name : VLAN60

Number of Filtered Groups : 2
Number of Standard Groups : 0
Number of Static Groups : 0
Total Multicast Groups Joined : 2

<table>
<thead>
<tr>
<th>Mode</th>
<th>EXCLUDE</th>
<th>INCLUDE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Filtered</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Standard</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Total</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>
```

show ip igmp statistics

Use the command to view the IGMP high-level statistics for all VLANs on the switch.

Syntax

show ip igmp statistics

Displaying statistics for IGMP joined groups

```
switch(config)# show ip igmp statistics

IGMP Service Statistics

Total VLAN's with IGMP enabled: 33
Current count of multicast groups joined: 21

IGMP Joined Group Statistics

<table>
<thead>
<tr>
<th>VLAN ID</th>
<th>VLAN Name</th>
<th>Total Groups</th>
<th>Filtered Groups</th>
<th>Standard Groups</th>
<th>Static Groups</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>DEFAULT_VLAN</td>
<td>52</td>
<td>50</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>22</td>
<td>VLAN-2</td>
<td>80</td>
<td>75</td>
<td>5</td>
<td>0</td>
</tr>
<tr>
<td>33</td>
<td>VLAN-3</td>
<td>1100</td>
<td>1000</td>
<td>99</td>
<td>1</td>
</tr>
</tbody>
</table>
```

show ip igmp vlan config

This command is used to show the IGMP configuration for a VLAN.

Syntax

show ip igmp vlan (vid) config

Example output
Below is the output when version is set to 3.

```
switch(vlan-60)# show ip igmp vlan 60 config

IGMP Service VLAN Config

VLAN ID : 60
VLAN NAME : VLAN60
IGMP Enabled : Yes
Querier Allowed [Yes] : No
Strict Mode : No
Last Member Query Interval (Seconds) [1] : 1
Querier Interval [125] : 125
Query Max. Response Time (Seconds) [10] : 10

<table>
<thead>
<tr>
<th>Port</th>
<th>Type</th>
<th>Port Mode</th>
<th>Forced Fast Leave</th>
<th>Fast Leave</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1000T</td>
<td>Auto</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>2</td>
<td>1000T</td>
<td>Auto</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>23</td>
<td>1000T</td>
<td>Auto</td>
<td>No</td>
<td>Yes</td>
</tr>
</tbody>
</table>
```

**show ip igmp config**

This command is used to show the global IGMP configuration.

**Syntax**

```
show ip igmp config
```

**Example output**

Below is the output when version is set to 3.

```
switch(vlan-60)# show ip igmp config

IGMP Service Config

Control unknown multicast [Yes] : Yes
Forced fast leave timeout [0] : 4
Delayed flush timeout [0] : 0
Look-up Mode [mac] : ip

<table>
<thead>
<tr>
<th>VLAN ID</th>
<th>VLAN Name</th>
<th>IGMP Enabled</th>
<th>Querier Allowed</th>
<th>IGMP Version</th>
<th>Querier Interval</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>DEFAULT_VLAN</td>
<td>No</td>
<td>Yes</td>
<td>2</td>
<td>125</td>
</tr>
<tr>
<td>60</td>
<td>VLAN60</td>
<td>Yes</td>
<td>No</td>
<td>3</td>
<td>125</td>
</tr>
</tbody>
</table>
```

**show ip igmp vlan group**

**Syntax:**

```
show ip igmp vlan vid group ip-addr
```

**Example output**

Here is an example of Group information for a VLAN with a filtered address group

```
switch(config)# show ip igmp vlan 22 group 239.20.255.7

IGMP Service Protocol Group Info

VLAN ID: 22
VLAN NAME: VLAN-2
```
Filtered Group Address: 239.20.255.7
Last Reporter: 192.168.0.2
Up Time: 1 hr 14 min 5 sec

<table>
<thead>
<tr>
<th>Port</th>
<th>Port Type</th>
<th>Port Mode</th>
<th>Expires</th>
<th>Access</th>
</tr>
</thead>
<tbody>
<tr>
<td>A1</td>
<td>100/1000T</td>
<td>Auto</td>
<td>1hr 2min 5sec</td>
<td>Host</td>
</tr>
</tbody>
</table>

**igmp reload**

This command is used to reset the IGMP state on all interfaces.

**Syntax**

`igmp reload`

**Example output**

IGMP application is in Error State as System Resources are exhausted. Traffic will flood. Please disable IGMP on all VLANs or Issue the Command "igmp reload" to take it out of Error. Refer to your product manual for information on IGMP resource consumption.

This is the output for igmp reload
For introductory and general information, see the sections beginning with PIM-DM. For information on PIMv6 DM, see the IPv6 Guide for your switch.

Overview of PIM-DM

This chapter describes protocol-independent multicast (PIM) routing operation on the switches covered in this guide and how to configure it with the switch's built-in interfaces. It is assumed that you have an understanding of multimedia traffic control with IP multicast (IGMP).

Global and PIM configuration contexts

---

**NOTE**

PIM-DM operation requires a routing protocol enabled on the routing switch. You can use RIP, OSPF, and/or static routing. The examples in this section use RIP.

---

Enabling or disabling IP multicast routing

**Syntax:**

```
ip multicast-routing
no ip multicast-routing
```

Enables or disables IP multicast routing on the routing switch. IP routing must be enabled before enabling multicast routing using the command `ip routing`.  
Default: Disabled

Enabling or disabling PIM at the global level; placing the CLI in the PIM context

**Syntax:**

```
router pim enable
no router pim
```

Enables or disables PIM at the global level and places the CLI in the PIM context. IP routing and IP multicast routing must first be enabled.  
Default: Disabled.

Setting the interval in seconds between successive state-refresh messages originated by the routing switch

**Syntax:**

```
router pim state-refresh [10-300]
```

The state-refresher timer cannot be set as it yet to be implemented, even though the CLI to set the timer is available. Only the routing switch connected directly to the unicast source initiates state-refresh packets. All other PIM routers in the network only propagate these state-refresh packets.  
Default: 60 seconds
Enabling and disabling PIM SNMP traps

Syntax:
```
router pim trap [[all] | neighbor-loss | hardware-mrt-full | software-mrt-full
no router pim trap [[all] | neighbor-loss | hardware-mrt-full | software-mrt-full]
```

Executed in the PIM context, this command enables and disables these PIM SNMP traps:

- **[all]**
  - Enable/disable all PIM notification traps.
- **[neighbor-loss]**
  - Enable/disable the notification trap sent when the timer for a multicast router neighbor expires and the switch has no other multicast router neighbors on the same VLAN with a lower IP address.
  - Default: Disabled
- **[hardware-mrt-full]**
  - Enable/disable notification trap when the hardware MRT is full (2046 active flows.) In this state, any additional flows are handled by the software MRT, which increases processing time for the affected flows.
  - Default: Disabled
- **[software-mrt-full]**
  - Enable/disable notification trap when the routing switch's software MRT is full (routing resources for active flows are exhausted.)
  - Default: Disabled

In this state, the routing switch does not accept any additional flows.

Configuring PIM in the Global and PIM context

In State-refresh packets and bandwidth conservation, the "#1" routing switch is directly connected to the multicast sources for the network. For this example, suppose that you are choosing the following:

- Reduce the state-refresh time from the default 60 seconds to 30 seconds. (The routing switch transmits state-refresh packets only if it is directly connected to the multicast source.)
- Configure an SNMP trap to notify your network management station if the routing switch's hardware multicast routing table becomes filled to the maximum of 2046 active flows.

To configure global-level PIM operation for the "#1" routing switch, you would use the commands shown in the following figure.

Configuring PIM-DM on a routing switch at the global level
After configuring the global-level PIM operation on a routing switch, go to the device’s VLAN context level for each VLAN you want to include in your multicast routing domain. For more information, see Viewing the current configuration for the specified VLAN (PIM interface).

### PIM VLAN (interface) configuration context

#### Enabling multicast routing on the VLAN interface to which the CLI is currently set

**Syntax:**

```
ip pim-dense
no ip pim-dense
```

- **Set the IP address in the VLAN prior to configuring PIM.**

```
vlan [vid] ip pim-dense
no vlan [vid] ip pim-dense
```

- Enables multicast routing on the VLAN interface to which the CLI is currently set. The no form disables PIM on the VLAN.

- **Default:** Disabled

- **Specifying the IP address to use as the source address for PIM protocol packets outbound on the VLAN**
Syntax:
no ip pim-dense [ip-addr any | sourceip-address]
no vlan[vid]ip pim-dense [ip-addr | any | sourceip-address]

In networks using multinetted VLANs, all routers on a given VLAN intended to route multicast packets must have a least one common subnet on that VLAN. Use this command when the VLAN is configured with multiple IP addresses (multinetting) to specify the IP address to use as the source address for PIM protocol packets outbound on the VLAN.

- Use ip-address to designate a single subnet in cases where multicast routers on the same multinetted VLAN are not configured with identical sets of subnet IP addresses.
- Use all if the multinetted VLAN is configured with the same set of subnet addresses.

Default: the primary VLAN

Changing the frequency at which the routing switch transmits PIM hello messages on the current VLAN

Syntax:
ip pim-dense [hello-interval 5-30]
  vlan [vid]ip pim-dense [hello-interval 5-30]

Changes the frequency at which the routing switch transmit PIM hello messages on the current VLAN. The routing switch uses hello packets to inform neighboring routers of its presence. The routing switch also uses this setting to compute the hello hold time, which is included in hello packets sent to neighbor routers. hello hold time tells neighbor routers how long to wait for the next hello packet from the routing switch. If another packet does not arrive within that time, the router removes the neighbor adjacency on that VLAN from the routing table, which removes any flows running on that interface.

Shortening the hello interval reduces the hello hold time. This has the effect of changing how quickly other routers will stop sending traffic to the routing switch if they do not receive a new hello packet when expected.

**NOTE**

Not used with the noform of the ip pim-dense command.

Example:
If multiple routers are connected to the same VLAN and the routing switch requests multicast traffic, all routers on the VLAN receive that traffic. (Those that have pruned the traffic will drop it when they receive it.)

If the upstream router loses contact with the routing switch receiving the multicast traffic (that is, fails to receive a hello packet when expected), the shorter hello interval causes it to stop transmitting multicast traffic onto the VLAN sooner, resulting in less unnecessary bandwidth usage.

Changing maximum time before the routing switch transmits the initial PIM hello message on the VLAN

Syntax:
ip pim-dense [hello-delay 0-5]
  vlan [vid]ip pim-dense [hello-delay 0-5]

Changes the maximum time in seconds before the routing switch transmits the initial PIM hello message on the current VLAN. In cases where a new VLAN activates with connections to multiple routers, if all of the connected routers sent hello packets at the same time, the receiving router could become momentarily overloaded. This value randomizes the transmission delay to a time between 0 and the hello delay setting. Using 0 means no delay.

After the routing switch sends the initial hello packet to a newly detected VLAN interface, it sends subsequent hello packets according to the current hello interval setting.
Changing the interval the routing switch waits for the graft ack from another router before resending the graft request

Syntax:

```
ip pim-dense [graft-retry-interval[1-10]]
```
```
vlan[vid]ip pim-dense [graft-retry-interval[1-10]]
```

Graft packets result when a downstream router transmits a request to join a flow. The upstream router responds with a graft acknowledgment packet. If the graft ack (acknowledgement) is not received within the time period of the graft-retry-interval, it resends the graft packet. The command `[graft-retry-interval[1-10]]` changes the interval (in seconds) the routing switch waits for the graft ack from another router before resending the graft request.

Default: 5 seconds

Changing the number of times the routing switch retries sending the same graft packet to join a flow

Syntax:

```
ip pim-dense [max-graft-retries[1-10]]
```
```
vlan[vid]ip pim-dense [max-graft-retries[1-10]]
```

Changes to the number of times the routing switch will retry sending the same graft packet to join a flow. If a graft ack response is not received after the specified number of retries, the routing switch ceases trying to join the flow. In this case the flow is removed until either a state-refresh from upstream re-initiates the flow or an upstream router floods the flow.

Increasing this value helps to improve multicast reliability.

Default: 3 attempts

Enabling the LAN prune delay option on the current VLAN

Syntax:

```
ip pim-dense [lan-prune-delay]
```
```
vlan[vid]ip pim-dense [lan-prune-delay]
```

Enables the LAN prune delay option on the current VLAN. With lan-prune-delay enabled, the routing switch informs downstream neighbors how long it will wait before pruning a flow after receiving a prune request. Other, downstream routers on the same VLAN must send a Join request to override the prune before the lan-prune-delay times out if they want the flow to continue. This prompts any downstream neighbors with hosts continuing to belong to the flow to reply with a Join. If no joins are received after the LAN prune-delay period, the routing switch prunes the flow.

The propagation-delay and override-interval settings determine the lan-prune-delay setting.

Default: Enabled
Computing the lan-prune-delay setting

Syntax:

```
ip pim-dense [propagation-delay[250-2000]]
ip pim-dense [override-interval[500-6000]]
vlans[vid]ip pim-dense [override-interval[500-6000]]
```

A routing switch sharing a VLAN with other multicast routers uses these two values to compute the lan-prune-delay setting for how long to wait for a PIM-DM Join after receiving a prune packet from downstream for a particular multicast group.

Defaults: propagation-delay=500 milliseconds; override-interval = 2500 milliseconds

Upstream router prune

A network may have multiple routing switches sharing VLAN "X". When an upstream routing switch initially floods traffic from multicast group "X" to VLAN "Y", if one of the routing switches on VLAN "Y" does not want this traffic, it issues a prune response to the upstream neighbor. The upstream neighbor then goes into a prune pending state for group "X" on VLAN "Y". (During this period, the upstream neighbor continues to forward the traffic.)

During the prune pending period, another routing switch on VLAN "Y" can send a group "X" Join to the upstream neighbor. If this happens, the upstream neighbor drops the prune pending state and continues forwarding the traffic. If no routers on the VLAN send a Join, the upstream router prunes group "X" from VLAN "Y" when the lan-prune-delay timer expires.

Setting the multicast datagram time-to-live (router hop-count) threshold for the VLAN

Syntax:

```
ip pim-dense [ttl-threshold[0-255]]
vlans[vid]ip pim-dense [ttl-threshold[0-255]]
```

Sets the multicast datagram time-to-live (router hop-count) threshold for the VLAN. Any IP multicast datagrams or state-refresh packets with a TTL less than this threshold will not be forwarded out the interface. The default value of 0 means all multicast packets are forwarded out the interface.

The VLAN connected to the multicast source does not receive state refresh packets and thus is not state-refresh capable. Downstream VLANs in the switches covered in this guide are state-refresh capable. This parameter provides a method for containing multicast traffic within a network, or even within specific areas of a network. Initially, the multicast traffic source sets a TTL value in the packets it transmits. Each time one of these packets passes through a multicast routing device, the TTL setting decrements by 1. If the packet arrives with a TTL lower than the mroutettl-threshold, the routing switch does not forward the packet.

Changing this parameter on a routing switch requires knowledge of the TTL setting of incoming multicast packets:

- A value that is too high can allow multicast traffic to go beyond your internal network.
- A value that is too low may prevent some intended hosts from receiving the desired multicast traffic.

Default: 0--forwards multicast traffic regardless of packet TTL setting

Example of configuring PIM-DM operation at the VLAN level

The network in the following figure uses VLAN 25 for multicast traffic. However, this VLAN is multinetted and there is only one subnet (10.38.10.x) in VLAN 25 that is common to all three routing switches. Thus, when configuring VLAN 25 on these routing switches to perform multicast routing, it is necessary to use ip pim-dense source-ip-address to designate the common subnet as the source address for outbound multicast traffic on VLAN
25. (If only identical subnets were present in the multinetted VLAN 25 configuration on all three devices, the ip pim-dense ip-addr any command would be used instead.) The other VLANs in the network are not multinetted and therefore do not require the ip pim-dense ip-addr any|source-ip-address option.

For this example, assume that the VLANs and IP addressing are already configured on the routing switch.

**Figure 3  Multicast network with a multinetted VLAN**

![Multicast network with a multinetted VLAN](image)

The preceding figure illustrates the steps for configuring multicast routing at the VLAN level for the switch #1.

```
switch(config)# vlan 25
switch(vlan-25)# ip igmp
switch(vlan-25)# ip rip
switch(vlan-25)# ip pim-dense ip-addr 10.38.10.1
switch(vlan-25-pim-dense)# vlan 27
switch(vlan-27)# ip igmp
switch(vlan-27)# ip rip
switch(vlan-27)# ip pim-dense
switch(vlan-27-pim-dense)# vlan 29
switch(vlan-29)# ip igmp
switch(vlan-29)# ip rip
switch(vlan-29)# ip pim-dense
switch(vlan-29-pim-dense)# write mem
switch(vlan-29-pim-dense)# exit
switch(vlan-29)# exit
```
Figure 4  Multicast routing configuration on switch #1 in Multicast network with a multinetted VLAN

```
Switch(config)# show run
...
ip routing
ip multicast-routing ...
vlan 25
   name "VLAN25"
   untagged A11-A15,A17
   ip address 10.29.30.1 255.255.255.0
   ip igmp
   exit
vlan 25
   name "VLAN25"
   [untagged A20-A24 \[\]
   ip address 10.36.10.1 255.255.255.0
   ip address 10.36.11.1 255.255.255.0
   ip igmp
   exit
vlan 27
   name "VLAN27"
   untagged A6-A10,A18
   ip address 10.27.30.1 255.255.255.0
   ip igmp
   exit
   [router rip
   enable \[\]
   exit
   [router pim
   enable \[state-refresh 45
trap hardware-mrt-full\]
   exit
   [ip rip 10.38.10.1 ...
   [ip rip 10.38.11.1 ...
   [ip pim-dense ...
   ip address 10.38.10.1 ...
   [untagged <port>
   ip address <ip address> <subnet mask>
   ip rip <ip address>
   ip pim-dense
   ip addr any
   exit
   [untagged <port>
   ip address <ip address> <subnet mask>
   ip rip <ip address>
   ip pim-dense
   ip addr any
   exit
```

Displaying PIM data and configuration settings

Displaying PIM route data

**Syntax:**
```
show ip [mroute]
```

Without parameters, lists multicast route entries in the following situations:

- When the PIM-DM router is actively forwarding a multicast flow out an interface (VLAN.)
- On a PIM-DM originator router (source directly connected) when traffic is entering the router but not forwarding
The neighbor field will be empty in this case.

- On a PIM-DM Non-originator router for a short duration after a flow’s initial flood/prune cycle is seen. This entry is cleared after 5 minutes unless the flow is connected within that time period.

**[Group Address]**
The multicast group IP address for the specific flow (source-group pair.)

**[Source Address]**
The unicast address of the flow’s source.

**[neighbor]**
The IP address of the upstream multicast router interface (VLAN) from which the multicast flow is coming. A blank field indicates that the multicast source is directly connected to the router.

**[VLAN]**
The interface on which the router receives the multicast flow.

**Showing the route entry data on the “#2” routing switch**
The next figure displays the `show ip mroute` output on the “#2” routing switch shown in Example of configuring PIM-DM operation at the VLAN level. This case illustrates two multicast groups from the same multicast source.

```
switch(config)# show ip mroute
IP Multicast Route Entries
Total number of entries : 2
Group Address Source Address Neighbor VLAN
----------------- --------------- --------------- ----
239.255.255.1 10.27.30.2 10.29.30.1 29
239.255.255.5 10.27.30.2 10.29.30.1 29
```

**Displays the PIM interfaces currently configured**

**Syntax**
```
show ip [mroute] [interface vid]
```
Lists the PIM interfaces (VLANs) currently configured in the routing switch.

- **VLAN**: Lists the VID of each VLAN configured on the switch to support PIM-DM.
- **IP Address**: Lists the IP addresses of the PIM interfaces (VLANs.)
- **Mode**: Shows dense only.

**Output for routing switch “#1”**
```
switch(config)#show ip mroute interface
PIM Interfaces

<table>
<thead>
<tr>
<th>VLAN</th>
<th>IP Address</th>
<th>Mode</th>
</tr>
</thead>
<tbody>
<tr>
<td>102</td>
<td>102.1.1.3</td>
<td>sparse</td>
</tr>
<tr>
<td>103</td>
<td>103.1.1.3</td>
<td>sparse</td>
</tr>
</tbody>
</table>
```

**Viewing VLAN, protocol identity, and TTL settings**

**Syntax:**
```
show ip [mroute] [interface vid]
```
The show ip mroute interface command on routing switch "#2" in Multicast network with a multinetted VLAN

```
switch(config)# show ip mroute interface 29
  IP Multicast Interface
  VLAN    : 29
  Protocol : PIM-DM
  TTL Threshold : 0
```

Viewing data for a specified flow (multicast group)

**Syntax:**
```
show ip [mroute] [multicast-ip-addrsource-ip-addr]
```

Lists the following data for the specified multicast flow (source-group pair):

**[Group Address]**
The multicast group IP address for the specified flow.

**[Source Address]**
The source IP address for the specified flow.

**[neighbor]**
Lists the IP address of the upstream next-hop router running PIM-DM; that is, the router from which the routing switch is receiving datagrams for the current multicast group.

This value is 0.0.0.0 if the routing switch has not detected the upstream next-hop router's IP address. This field is empty if the multicast server is directly connected to the routing switch.

**[VLAN]**
The interface on which the router receives the multicast flow.

**[Up Time (Sec)]**
The elapsed time in seconds since the routing switch learned the information for the current instance of the indicated multicast flow.

---

**[Note]**
On an originator router, when a forwarding flow moves to a non-forwarding state (i.e. when pruned) the Up Time value for that flow is reset to 0.

**[Expire Time (Sec)]**
An mroute which is in a forwarding state – one which represents an active, connected flow for which there are downstream routers and/or locally connected hosts interested in the flow – does not expire. When other PIM-DM routers or locally connected hosts are no longer interested in an active flow, the related mroute on an originator router moves to a blocking state, and an mroute in this state does not expire either. In both cases the mroute is only removed by the system when it is no longer needed and so the displayed value for expire time in these situations is not meaningful.

For an mroute on an originator router whose flow is no longer active - including mroutes on non-originators whose flow has been pruned – expire time indicates when the mroute entry will eventually be cleared.

**Multicast Routing Protocol**
Identifies the multicast routing protocol through which the current flow was learned.

**Unicast Routing Protocol**
Identifies the IP routing protocol through which the routing switch learned the upstream interface for the current multicast flow. The listed protocol will be one of the following:
RIP
connected
OSPF
static route
other

Metric
Indicates the path cost upstream to the multicast source. Used when multiple multicast routers contend to determine the best path to the multicast source. The lower the value, the better the path.

Metric Pref
Used when multiple multicast routers contend to determine the path to the multicast source. When this value differs between routers, PIM selects the router with the lowest value. If Metric Pref is the same between contending multicast routers, then PIM selects the router with the lowest Metric value to provide the path for the specified multicast traffic. (Different vendors assign differing values for this setting.)

Asset Timer
The time remaining until the router ceases to wait for a response from another multicast router to negotiate the best path back to the multicast source. If this timer expires without a response from any contending multicast routers, then the router assumes it is the best path, and the specified multicast group traffic will flow through the router.

RP Tree
This field is not relevant to PIM-DM and will always display No.

Downstream interfaces
For each downstream interface the following information is shown:

[VLAN]
Lists the [VID]
of the VLAN that the routing switch is using to send the outbound packets of the current multicast flow to the next-hop router.

[state]
Indicates whether the outbound VLAN and next-hop router for the current multicast flow are receiving datagrams.

Pruned
The routing switch has not detected any joins from the current multicast flow and is not currently forwarding datagrams in the current VLAN.

Forwarding
The routing switch has received a join for the current multicast flow and is forwarding datagrams in the current VLAN.

Up Time (Sec)
The natural state of a downstream interface in PIM-DM is to forward multicast flows and DM will flood a new flow out all interfaces on a router where there are connected PIM-DM neighbors and/or joined hosts. If there are ultimately no receivers for the flow downstream, the flow will be pruned back to the originator router. This prune state is maintained on all PIM-DM routers by state refresh message sends by the originator and corresponding prune replies from downstream routers. However if a prune reply is not received (i.e. there is now a receiver), expire time indicates how long before the interface will return to a forwarding state.

Expire Time (sec)
The natural state of a downstream interface in PIM-DM is to forward multicast flows and DM will flood a new flow out all interfaces on a router where there are connected PIM-DM neighbors and/or joined hosts. If there are ultimately no receivers for the flow downstream, the flow will be pruned back to the originator router. This prune
state is maintained on all PIM-DM routers by state refresh message sends by the originator and corresponding prune replies from downstream routers. However if a prune reply is not received (i.e. there is now a receiver), expire time indicates how long before the interface will return to a forwarding state.

**Output for routing switch "#1" in Multicast network with a multinetted VLAN**

A populated neighbor field indicates that the multicast server is directly connected to the routing switch (neighbor field is highlighted in bold below.)

```
switch(config)# show ip mroute 239.255.255.5 10.27.30.2
IP Multicast Route Entry
  Group Address : 239.255.255.5
  Source Address : 10.27.30.2
  Source Mask : 255.255.255.0
  Neighbor : 10.30.229.1
  VLAN : 27
  Up time (sec) : 408
  Expire Time (sec) : 150
  Multicast Routing Protocol : PIM-DM
  Unicast Routing Protocol : rip

  Downstream Interfaces
  VLAN State Up time (sec) Expire Time (sec)
  ------- ---------- ---------------
  28 pruned 408 98
```

**Show IP PIM**

**Syntax**
```
show ip pim neighbors
```
Show PIM protocol operational and configuration information.

**neighbors**
Show PIM neighbor information.

**show ip pim**

```
PIM Global Parameters
  PIM Status : Enabled
  State Refresh Interval (sec) : 60
  Join/Prune Interval (sec) : 60
  SPT Threshold : Enabled
  Traps : none
```

**show ip pim neighbors**
Lists PIM neighbor information for all PIM neighbors connected to the routing switch.

**neighbor**
Show PIM neighbor information.

**IP Address**
Lists the IP address of a neighbor multicast router.

**VLAN**
Lists the VLAN through which the routing switch connects to the indicated neighbor.

**Up Time**
Shows the elapsed time during which the neighbor has maintained a PIM route to the routing switch.

**Expire Time**
Indicates how long before the router will age-out a PIM neighbor/adjacency relationship on the specified interface (VLAN.) When a neighbor/adjacency expires and that neighbor was the last one on the interface, multicast data and state refresh packets will no longer be sent out that interface. Receipt of a periodic PIM hello message from the neighboring PIM router resets this timer to the hold time value stored in the hello message. If the ip-addr is specified then detailed information for the specified neighbor is shown.

<table>
<thead>
<tr>
<th>IM Neighbors</th>
<th>VLAN Up Time (sec)</th>
<th>Expire Time (sec)</th>
</tr>
</thead>
<tbody>
<tr>
<td>20.1.1.2</td>
<td>20</td>
<td>193</td>
</tr>
</tbody>
</table>

### Listing the PIM interfaces (VLANs) currently configured in the routing switch

**Syntax:**

```text
show ip pim [interface]
```

Lists the PIM interfaces (VLANs) currently configured in the routing switch.

**[VLAN]**

Lists the VID of each VLAN configured on the switch to support PIM-DM.

**[ip address]**

Lists the IP addresses of the PIM interfaces (VLANs.)

**[mode]**

Shows dense only.

#### Output for routing switch "#1" in Multicast network with a multinetted VLAN

```text
switch(config)# show ip pim interface
PIM Interfaces
<table>
<thead>
<tr>
<th>VLAN IP Address</th>
<th>Mode</th>
</tr>
</thead>
<tbody>
<tr>
<td>25 10.38.10.1</td>
<td>dense</td>
</tr>
<tr>
<td>27 10.27.30.1</td>
<td>dense</td>
</tr>
<tr>
<td>29 10.29.30.1</td>
<td>dense</td>
</tr>
</tbody>
</table>
```

### Viewing the current configuration for the specified VLAN (PIM interface)

**Syntax:**

```text
show ip pim [interface [vid]]
```

Displays the current configuration for the specified VLAN (PIM interface.)

#### Output for routing switch "#1" in Multicast network with a multinetted VLAN

```text
switch(config)# show ip pim interface 29
PIM Interface
| VLAN : 29 |
| IP Address : 10.29.30.1 |
| Mode : dense |
| Hello Interval (sec) : 30 |
| Hello Delay (sec) : 5 |
| Graft Retry Interval (sec) : 3 |
| Max Graft Retries : 2 |
| Override Interval (msec) : 2500 |
| Lan Prune Delay : Yes |
| Propagation Delay (msec) : 500 |
| Lan Delay Enabled : No |
| SR TTL Threshold : 2 |
| State Refresh Capable : No |
```
### Table 8: PIM interface configuration settings

<table>
<thead>
<tr>
<th>Field</th>
<th>Default</th>
<th>Control command</th>
</tr>
</thead>
<tbody>
<tr>
<td>VLAN</td>
<td>N/A</td>
<td><code>vlan vid ip pim-dense</code></td>
</tr>
<tr>
<td>IP</td>
<td>N/A</td>
<td>`vlan vid ip pim-dense any</td>
</tr>
<tr>
<td>Mode</td>
<td>dense</td>
<td>PIM-dense or PIM-sparse</td>
</tr>
<tr>
<td>Hello interval (sec)</td>
<td>30</td>
<td><code>ip pim-dense hello interval 5 - 30</code></td>
</tr>
<tr>
<td>Hello hold time</td>
<td>105</td>
<td>The routing switch computes this value from the current hello interval and includes it in the hello packets the routing switch sends to neighbor routers. Neighbor routers use this value to determine how long to wait for another hello packet from the routing switch. See the description of the hello interval on PIM VLAN (interface) configuration context.</td>
</tr>
<tr>
<td>Hello delay</td>
<td>5</td>
<td><code>vlan vid ip pim-dense hello delay 0 - 5</code></td>
</tr>
<tr>
<td>Graft retry interval (sec)</td>
<td>3</td>
<td><code>vlan vid ip pim-dense graft-retry-interval 1 - 10</code></td>
</tr>
<tr>
<td>Max graft retries</td>
<td>2</td>
<td><code>vlan vid ip pim-dense graft-retries 1 - 10</code></td>
</tr>
<tr>
<td>Override interval (msec)</td>
<td>2500</td>
<td><code>vlan vid ip pim-dense override-interval 500 - 6000</code></td>
</tr>
<tr>
<td>Propagation delay (msec)</td>
<td>500</td>
<td><code>vlan vid ip pim-dense propagation-delay 250-2000</code></td>
</tr>
<tr>
<td>SR TTL threshold (router hops)</td>
<td>0</td>
<td><code>vlan vid ip pim-dense ttl-threshold 0 - 255</code></td>
</tr>
<tr>
<td>LAN prune delay</td>
<td>Yes</td>
<td><code>vlan vid ip pim-dense lan-prune-delay</code></td>
</tr>
<tr>
<td>LAN delay enabled</td>
<td>No</td>
<td>Shows [Yes] if all multicast routers on the current VLAN interface enabled LAN-prune-delay. Otherwise, shows No</td>
</tr>
<tr>
<td>State-refresh capable</td>
<td>N/A</td>
<td>Indicates whether the VLAN responds to state-refresh packets. The VLAN connected to the multicast source does not receive state-refresh packets and thus is not state-refresh capable. Downstream VLANs in the switches are state-refresh capable.</td>
</tr>
</tbody>
</table>

### Viewing PIM-specific information from the IP multicast routing table (MRT)
Syntax:
show ip pim [mroute]

This command displays exactly the same output as the command
show ip [mroute]

Viewing the PIM route entry information for the specified multicast group (flow)

Syntax:
show ip pim [mroute [multicast-group-address multicast-source-address]]
[Group Address]
Lists the specified multicast group address.
[Source Address]
Lists the specified multicast source address.
[Source Mask]
Lists the network mask for the multicast source address.
Metric
Indicates the path cost upstream to the multicast source. Used when multiple multicast routers contend to
determine the best path to the multicast source. The lower the value, the better the path.
Metric Pref
Used when multiple multicast routers contend to determine the path to the multicast source. When this value
differs between routers, PIM selects the router with the lowest value.
If Metric Pref is the same between contending multicast routers, PIM selects the router with the lowest Metric
value to provide the path for the specified multicast traffic. (Different vendors assign differing values for this
setting.)
Assert Timer
The time remaining until the routing switch ceases to wait for a response from another multicast router to
negotiate the best path back to the multicast source. If this timer expires without a response from any contending
multicast routers, the routing switch assumes it is the best path, and the specified multicast group traffic will flow
through the routing switch.
Downstream Interfaces
[VLAN]
Lists the VID of the destination VLAN on the next-hop multicast router.
Prune Reason
Identifies the reason for pruning the flow to the indicated VLAN:
Prune
A neighbor multicast router has sent a prune request.
Assert
Another multicast router connected to the same VLAN has been elected to provide the path for the specified
multicast group traffic.
Other
Used where the VLAN is in the pruned state for any reason other than the above two reasons (such as no
neighbors exist and no directly connected hosts have done joins.)

Routing switch "#1" in Multicast network with a multinetted VLAN showing a multicast
group from a directly connected source
switch(config)# show ip pim mroute 239.255.255.1 10.27.30.2
PIM Route Entry
  Group Address : 239.255.255.1
  Source Address : 10.27.30.2
  Source Mask : 255.255.255.0
  Metric : 3
  Metric Pref : 120
  Assert Timer : 0
Downstream Interfaces
  VLAN Prune Reason
  ---- ----------
  28 prune

Listing PIM neighbor information for all PIM neighbors connected to the routing switch

Syntax:
show ip pim [neighbor]

IP Address
Lists the IP address of a neighbor multicast router.

VLAN
Lists the VLAN through which the routing switch connects to the indicated neighbor.

Up Time
Shows the elapsed time during which the neighbor has maintained a PIM route to the routing switch.

Expire Time
Indicates how long before the routing switch ages-out the current flow (group membership.) This value decrements until:

- Reset by a state-refresh packet originating from the upstream multicast router. (The upstream multicast router issues state-refresh packets for the current group as long as it either continues to receive traffic for the current flow or receives state-refresh packets for the current flow from another upstream multicast router.
- Reset by a new flow for the current multicast group on the VLAN.
- The timer expires (reaches 0.) In this case, the switch has not received either a state-refresh packet or new traffic for the current multicast group and ages-out (drops) the group entry.

If the ip-addr is specified, detailed information for the specified neighbor is shown.

PIM neighbor output
This example simulates output from routing switch "#1" in Example of configuring PIM-DM operation at the VLAN level. The data identifies the first downstream neighbor ("routing switch #2").

switch(config)# show ip pim neighbor
PIM Neighbors
  IP Address  VLAN  Up Time (sec)  Expire Time (sec)
     --------    ----   ------------  ------------
  10.29.30.2  29   196           89

Syntax:
show ip pim [neighbor [ip-address]]
Lists the same information as the show ip pim neighbor

Showing a specific neighbor
This example simulates output from routing switch "#1" in Example of configuring PIM-DM operation at the VLAN level. The data is from the first downstream neighbor (routing switch "#2").
PIM-DM

In a network where IP multicast traffic is transmitted for multimedia applications, traffic is blocked at routed interface (VLAN) boundaries unless a multicast routing protocol is running. PIM is a family of routing protocols that form multicast trees to forward traffic from multicast sources to subnets are using a protocol such as IGMP to request the traffic. PIM relies on the unicast routing tables created by any of several unicast routing protocols to identify the path back to a multicast source, known as reverse path forwarding (RPF.) Based on information provided by the unicast routing tables, PIM sets up a distribution tree for the multicast traffic. The PIM-DM and PIM-SM protocols on the switches enable and control multicast traffic routing.

IGMP provides the multicast traffic link between a host and a multicast router running PIM-DM or PIM-SM. IGMP and either PIM-DM or PIM-SM must be enabled on VLANs whose member ports have directly connected hosts with a valid need to join multicast groups. PIM-DM is used in networks where, at any given time, multicast group members exist in relatively large numbers and are present in most subnets.

PIM-DM features

PIM-DM features on switches covered by this guide include:

Routing protocol support

PIM uses whichever unicast routing protocol is running on the routing switch. These can include:

- RIP
- OSPF
- Static routes
- Directly connected interfaces

VLAN interface support

The MRT supports up to 128 outbound VLANs at any given time. The sum of all outbound VLANs across all current flows on a router may not exceed 128. (A single flow may span one inbound VLAN and up to 128 outbound VLANs, depending on the VLAN memberships of the hosts actively belonging to the flow.)

Flow capacity

Up to 2046 flows are supported in hardware across a maximum of 128 outbound VLANs. (A flow is composed of an IP source address and an IP multicast group address, regardless of the number of active hosts belonging to the multicast group at any given time.)

IGMP compatibility

PIM-DM is compatible with IGMP (V1 to V3) and is fully interoperable with IGMP for determining multicast flows.

VRRP

PIM-DM is fully interoperable with VRRP to quickly transition multicast routes in the event of a failover.

MIB support

With some exceptions, PIM-DM supports the parts of the multicast routing MIB applicable to PIM-DM operation.

PIM draft specifications

Compatible with PIM-DM draft specifications (V1 and V2.)
PIM-DM operation

PIM-DM operates at the router level to direct traffic for a particular multicast group along the most efficient path to the VLANs which have hosts that have joined that group. A unicast source address and a multicast group address comprise a given source/group (S/G) pair. Multicast traffic moving from a source to a multicast group address creates a flow to the area(s) of the network requiring the traffic. The flow destination is the multicast group address and not a specific host or VLAN. A single multicast flow has one source and one multicast group address (destination), but may reach many hosts in different subnets, depending on which hosts have issued joins for the same multicast group.

PIM routes the multicast traffic for a particular S/G pair on paths between the source unicast address and the VLANs where it is requested (by joins from hosts connected to those VLANs.) Physical destinations for a particular multicast group can be hosts in different VLANs or networks. Individual hosts use IGMP configured per-VLAN to send joins requesting membership in a particular multicast group. All hosts that have joined a given multicast group (defined by a multicast address) remain in that group as long as they continue to issue periodic joins.

PIM-DM interoperates with IGMP and the switch's routing protocols for the switches covered by this guide. The PIM operates independently of the routing protocol you choose to run on your switches. This means that you can use PIM-DM with RIP, OSPF, or static routes configured. PIM-DM uses a unicast routing table to find the path to the originator of the multicast traffic and sets up multicast trees for distributing multicast traffic. This routing method is known as reverse path forwarding (RPF.)

For the flow of a given multicast group, PIM-DM creates a tree structure between the source and the VLANs where hosts have joined the group as shown in the following figure. The tree structure consists of:

- Extended branches to VLANs with hosts that currently belong to the group.
- Pruned branches to VLANs with no hosts that belong to the group.

Example of multicast tree for a given flow

![Multicast Tree Diagram]

When the routing switch detects a new multicast flow, it initially floods the traffic throughout the PIM-DM domain, then it prunes the traffic on the branches (network paths) where joins have not been received from individual hosts. This creates the tree structure shown in the preceding figure. The routing switch maintains individual branches in the multicast tree as long as there is at least one host maintaining a membership in the multicast group. When all of the hosts in a particular VLAN drop out of the group, PIM-DM prunes that VLAN from the
multicast tree. Similarly, if the routing switch detects a join from a host in a pruned VLAN, it adds that branch back into the tree.

Where the multicast routers in a network use one or more multinetted VLANs, there must be at least one subnet common to all routers on the VLAN. This is necessary to provide a continuous forwarding path for the multicast traffic on the VLAN. See PIM VLAN (interface) configuration context.

**Multicast flow management**

This section provides details on how the routing switch manages forwarding and pruned flows. This information is useful when you plan topologies to include multicast support and when viewing and interpreting the show command output for PIM-DM features.

**Initial flood and prune**

When a router running PIM-DM receives a new multicast flow, it initially floods the traffic to all downstream multicast routers. PIM-DM then prunes the traffic on paths to VLANs that have no host joins for that multicast address. (PIM-DM does not re-forward traffic back to its source VLAN.)

**Maintaining the prune state**

For a multicast group "X" on a given VLAN, when the last host belonging to group "X" leaves the group, PIM places that VLAN in a prune state; this means that the group "X" multicast traffic is blocked to that VLAN. The prune state remains until a host on the same VLAN issues a join for group "X", in which case the router cancels the prune state and changes the flow to the forwarding state.

**State-refresh packets and bandwidth conservation**

A multicast switch, if directly connected to a multicast source (such as a video conference application), periodically transmit state-refresh packets to downstream multicast routers. On routers that have pruned the multicast flow, the state-refresh packets keep the pruned state alive. On routers that have been added to the network after the initial flooding and pruning of a multicast group, the state-refresh packets inform the newly added router of the current state of that branch. This means that if all multicast routers in a network support the state-refresh packet, the multicast router directly connected to the multicast source performs only one flood-prune cycle to the edge of the network when a new flow (multicast group) is introduced and preserves bandwidth for other uses.

Some vendors' multicast routers do not offer the state-refresh feature. In this case, PIM-DM must periodically advertise an active multicast group to these devices by repeating the flood/prune cycle on the paths to such routers. For better traffic management in multicast-intensive networks where some multicast routers do not offer the state-refresh feature, you may want to group such routers where the increased bandwidth usage will have the least effect on overall network performance.

*Bandwidth conservation in switches with PIM-DM state-refresh*
### General configuration elements

PM-DM requires you to configure the following elements:

- **IP routing enabled on all routing switches you want to carry routed multicast traffic.**
- **Routing methods needed to reach the interfaces (VLANs) on which you want multicast traffic available for hosts in your network:**
  - Enable RIP or OSPF at both the global and VLAN levels on the routers where there are connected hosts that may issue multicast joins.
  - Configure static routes to and from the destination subnets.
- **Enable IP multicast routing.**
- **Enable IGMP on each VLAN when that VLAN has hosts that you want to join multicast groups. Repeat this action on every switch and router belonging to the VLAN.**
- **Enable PIM-DM at the global level on the routing switch and on the VLANs where you want to allow routed multicast traffic.**

---

When you initially enable PIM-DM, it is recommended that you leave the PIM-DM configuration parameters at their default settings. From the default, you can assess performance and make configuration changes when needed.

### Configuring PIM-DM

PIM-DM requires configuration on both the global level and on the VLAN (interface) level. The recommended configuration order is:

1. Enable IGMP on all VLANs where hosts may join a multicast group.
2. Enable the following at the global level:
   a. IP routing
   b. IP multicast routing
c. Router PIM and any non-default, global PIM settings you want to apply

d. Router RIP, Router OSPF, and/or a static route

3. If you selected RIP or OSPF in step 2: enable the same option on each VLAN where you want multicast routing to operate.

4. Enable the following in each VLAN context where you want multicast routing to operate:
   a. IP RIP or IP OSPF
   b. IP PIM
   c. Any non-default, VLAN-level IP PIM settings you want to apply

**PIM-DM DT**

PIM-DM DT is not VLAN specific. PIM-DM DT can be configured on any VLAN to enable PIM-DM DT. For more information, see Distributed Trunking chapter in the *Management and Configuration Guide* of your switch.

- When PIM-DM DT are enabled, the feature is in place.
- PIM-SM can be configured with DT.
- v1 modules must be disabled prior to PIM-DM DT. Module v1 is available with either feature individually but not when enabled together. Use the allow-v1-modules command to disable V1 modules. If PIM-DM DT is enabled first, a v1 module will fail.
- With v1 modules allowed, PIM and DT must be configured as a first configuration.
- In multiple VLANs, the configurations can be in combinations of DT and PIM-DM. The PIM-DM DT feature must be enabled on first combination of PIM-DM DT and disabled when the last pair is un-configured. There can be multiple such pairs but each is not strictly bound to the same VLAN.

For information on the no allow-v1-modules command, see the *Management and Configuration Guide* for your switch.

---

**show distributed-trunking consistency-parameters global**

**Syntax**

```
show distributed-trunking consistency-parameters global feature feature
```

Shows all the feature options available in command.

**dhcp-snooping**

Display DHCP snooping peer consistency details.

**igmp**

Display IGMP peer consistency details.

**loop-protect**

Display Loop protect peer consistency details.

---

**NOTE**

DT trunks can use jumbo VLAN as usual, but user needs to ensure that jumbo is configured on both the DT pairs, otherwise packet drops/fragmentations can be seen.
### mld
Display MLD peer consistency details.

### PIM-DM
Display PIM-DM peer consistency details.

#### Show distributed-trunking

<table>
<thead>
<tr>
<th></th>
<th>Local</th>
<th>Peer</th>
</tr>
</thead>
<tbody>
<tr>
<td>Image Version</td>
<td>K.16.01.0004x</td>
<td>K.16.01.0004x</td>
</tr>
<tr>
<td>IP Routing</td>
<td>Disabled</td>
<td>Disabled</td>
</tr>
<tr>
<td>Peer-keepalive interval</td>
<td>1000</td>
<td>1000</td>
</tr>
<tr>
<td>PIM-DM Support</td>
<td>Disabled</td>
<td>Disabled</td>
</tr>
<tr>
<td>PIM-SM Support</td>
<td>Disabled</td>
<td>Disabled</td>
</tr>
<tr>
<td>IGMP enabled VLANs on Local</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IGMP enabled VLANs on Peer</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PIM-DM-DT Enabled VLANs on Local</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PIM-DM-DT Enabled VLANs on Peer</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PIM-SM-DT Enabled VLANs on Local</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PIM-SM-DT Enabled VLANs on Peer</td>
<td></td>
<td></td>
</tr>
<tr>
<td>DHCP-snooping Enabled on Local</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>DHCP-snooping Enabled on Peer</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>DHCP-snooping Enabled VLANs on Local</td>
<td></td>
<td></td>
</tr>
<tr>
<td>DHCP-snooping Enabled VLANs on Peer</td>
<td></td>
<td></td>
</tr>
<tr>
<td>DHCP-snooping Max-Binding Configured on Local</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>DHCP-snooping Max-Binding Configured on Peer</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>DHCPv6-snooping Enabled on Local</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>DHCPv6-snooping Enabled on Peer</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>DHCPv6-snooping Enabled VLANs on Local</td>
<td></td>
<td></td>
</tr>
<tr>
<td>DHCPv6-snooping Enabled VLANs on Peer</td>
<td></td>
<td></td>
</tr>
<tr>
<td>DHCPv6-snooping Max-Binding Configured on Local</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>DHCPv6-snooping Max-Binding Configured on Peer</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>Loop-protect enabled VLANs on Local</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Loop-protect enabled VLANs on Peer</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MLD enabled VLANs on Local</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MLD enabled VLANs on Peer</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
show distributed-trunking consistency-parameters global

feature pim-dm

PIM-DM Enabled VLANs on Local : 20,30
PIM-DM Enabled VLANs on Peer : 20,30

Show distributed-trunking PIM-DM enabled

#show distributed-trunking consistency-parameters global

<table>
<thead>
<tr>
<th>Local</th>
<th>Peer</th>
</tr>
</thead>
<tbody>
<tr>
<td>-----</td>
<td>-----</td>
</tr>
<tr>
<td>Peer config unavailable.</td>
<td></td>
</tr>
<tr>
<td>Image Version</td>
<td>KB.15.16.0000x KB.15.16.0000x</td>
</tr>
<tr>
<td>IP Routing</td>
<td>Enabled Disabled</td>
</tr>
<tr>
<td>Peer-keepalive interval</td>
<td>1000 0</td>
</tr>
<tr>
<td>PIM-DM Support</td>
<td>Enabled Enabled</td>
</tr>
<tr>
<td>IGMP enabled VLANs on Local</td>
<td>:</td>
</tr>
<tr>
<td>IGMP enabled VLANs on Peer</td>
<td>:</td>
</tr>
<tr>
<td>PIM-DM enabled VLANs on Local</td>
<td>: 20,30</td>
</tr>
<tr>
<td>PIM-DM enabled VLANs on Peer</td>
<td>: 20,30</td>
</tr>
<tr>
<td>DHCP-snooping enabled VLANs on Local</td>
<td>:</td>
</tr>
<tr>
<td>DHCP-snooping enabled VLANs on Peer</td>
<td>:</td>
</tr>
<tr>
<td>Loop-protect enabled VLANs on Local</td>
<td>:</td>
</tr>
<tr>
<td>Loop-protect enabled VLANs on Peer</td>
<td>:</td>
</tr>
<tr>
<td>MLD enabled VLANs on Local</td>
<td>:</td>
</tr>
<tr>
<td>MLD enabled VLANs on Peer</td>
<td>:</td>
</tr>
</tbody>
</table>

Error Log

On configuring Distributed Trunking on a VLAN where PIM-SM is already configured the following errors may appear.

Table 9: Error Log

<table>
<thead>
<tr>
<th>Feature</th>
<th>Error</th>
<th>Message</th>
</tr>
</thead>
<tbody>
<tr>
<td>Configure DT when PIM enabled</td>
<td>SWITCH_ERRORMSG_DT_CANNOT_CONFIGURE_DT</td>
<td>Cannot configure Distributed Trunking on a VLAN port that has PIM-SM configured.</td>
</tr>
<tr>
<td></td>
<td>SWITCH_ERRORMSG_DT_CANNOT_CONFIGURE_DT_ON_PIM_DEFAULT_VLAN</td>
<td>Cannot configure Distributed Trunking because PIM-SM is enabled on DEFAULT VLAN.</td>
</tr>
<tr>
<td>Configure PIM-SM when DT enabled</td>
<td>SWITCH_ERRORMSG_DT_CANNOT_CONFIGURE_PIM</td>
<td>Cannot configure PIM-SM on a VLAN that has Distributed Trunking configured.</td>
</tr>
<tr>
<td>Feature</td>
<td>Error</td>
<td>Message</td>
</tr>
<tr>
<td>--------------------------------------</td>
<td>----------------------------------------------------------------------</td>
<td>-------------------------------------------------------------------------</td>
</tr>
<tr>
<td>V1 module with PIM-DM DT</td>
<td>Enabling V1 module with PIM-DM &amp; DT are enabled on the same VLAN.</td>
<td>Cannot enable V1 modules when PIM-DM and Distributed Trunking are configured on the same VLAN.</td>
</tr>
<tr>
<td></td>
<td>Enabling PIM-DM or DT when V1 module is allowed.</td>
<td>PIM-DM and Distributed Trunking cannot be configured on the same VLAN when V1 modules are enabled.</td>
</tr>
</tbody>
</table>

Distributed Trunking between different type of switches is not supported. If the switch platforms do not match, an error message will return similar to inconsistent criteria.

**Exception**

Distributed Trunking between different types of switches is not supported.

If the switch platforms do not match, an error message will return similar to inconsistent criteria.

**Operating notes**

**PIM-DM operating rules**

- The routing switch supports 2046 multicast flows in hardware. See, Flow capacity.
- The multicast routing table (MRT) that PIM-DM creates allows up to 128 outbound VLANs at any given time. PIM-DM supports multicast routing across 128 VLANs (16 for the 2930F switch).
- The routing switch allows one instance of PIM per VLAN. For networks using multinetted VLANs, all routers on the intended VLAN must have at least one common subnet if you intend on routing multicast packets. The routing switch provides a command for specifying which IP address PIM will use on each VLAN.
- For the 2930F switch, the maximum number of routes is limited to 200.

**PIM routers without state-refresh messaging capability**

A PIM router without a state-refresh messaging capability learns of currently active flows in a multicast network through periodic flood and prune cycles on the path back to the source. The switches covered in this guide sense downstream multicast routers that do not have the state-refresh capability and will periodically flood active multicast groups to these devices. This periodic flooding is not necessary if all downstream multicast routers are switches covered in this guide.

**Flow capacity**

The routing switch provides an ample multicast environment, supporting 2046 multicast flows in hardware across a maximum of 64 VLANs. (A flow comprises a unicast source address and a multicast group address, regardless of the number of active hosts belonging to the multicast group at any given time.)

**IGMP traffic high-priority disabled**
Enabling IP multicast routing to support PIM-DM operation has the effect of disabling IGMP traffic high-priority, if configured.

**ACLs and PIM**

With V2 modules, the outbound routed ACLs only match IP unicast addresses and not IP multicast addresses. When the chassis is in V3 only mode (no-allow-v2), outbound routed ACLs can also match IP multicast addresses.

**When to enable IGMP on a VLAN**

When PIM is enabled on a VLAN, it is not necessary to also enable IGMP unless there may be joins occurring on that VLAN. But if IGMP is enabled on a VLAN, you must also enable PIM if you want that VLAN to participate in multicast routing.

**IP address removed**

If you remove the IP address for a VLAN, the switch automatically removes the PIM configuration for that VLAN.

**Troubleshooting**

**Symptom: Noticeable slowdown in some multicast traffic**

If the switch is supporting more than 1022 active flows, this generates the message **Unable to learn HW IP multicast groups, table FULL** in the Event Log, because there is no room in the hardware MRT to add another multicast group. Software will route any multicast packets sent to multicast groups that are not in the hardware MRT, but it will be slower, and packets may be dropped if the data rate is greater than 3000 packets per second. See [Flow capacity](#).

---

**NOTE**

The PIM protocol uses one MRT entry for every IP multicast S/G pair that it is routing. An entry is not used if the multicast flow is bridged and not routed. Entries in this table are automatically aged-out if they are unused for a period of time.

---

**Heavy memory usage**

Heavy use of PIM (many S/G flows over many VLANs), combined with other memory-intensive features, can oversubscribe memory resources and impact overall performance. If available memory is exceeded, the switch drops any new multicast flows and generates appropriate Event Log messages. Corrective actions can include:

- Reducing the number of VLANs on the switches by moving some VLANs to another device.
- Freeing up system resources by disabling another, non-PIM feature.
- Moving some hosts to another device.

For more information, see [Operating notes](#) and [Messages related to PIM operation on page 66](#).

**IPv4 table operation**

The IPv4 table, which contains the active IP multicast addresses the switch is currently supporting, has 128k entries. However, the IPv4 table also contains IP host entries for every IP source or destination that the switch has learned, as well as ACL flow entries. Entries in this table are generally aged-out if they are unused for 5 minutes or more.
Messages related to PIM operation

These messages appear in the Event Log and, if syslog debug is configured, in the designated Debug destinations.

The \( \text{counter} \) value displayed at the end of each PIM Event Log message (and SNMP trap messages, if trap receivers are configured) indicates the number of times the switch has detected a recurring event since the last reboot. See the Management and Configuration Guide for your switch.

<table>
<thead>
<tr>
<th>Message</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>- alpha-string pkt, src IP ip-addr vid vlan-id (not a nbr) (counter)</td>
<td>A PIM packet arrived from another router for which no neighbor was found. May indicate a misconfiguration between the sending and receiving router. May also occur if a connected router is disconnected, then reconnected.</td>
</tr>
<tr>
<td>- Bad TTL in State Refresh pkt from IP source-ip-addr (counter)</td>
<td>The switch detected a TTL of 0 (zero) in the PIM portion of a state-refresh packet. (This is not the IP TTL.)</td>
</tr>
<tr>
<td>- Failed alloc of HW alpha-str for flow multicast-address, source-address (dup-msg-cnt)</td>
<td>There are more than 2046 active flows. The switch routes the excess through software, which processes traffic at a slower rate. If this will be an ongoing or chronic condition, transfer some of the flows to another router.</td>
</tr>
<tr>
<td>- Failed to alloc a PIM data-type pkt (counter)</td>
<td>The router was unable to allocate memory for a PIM control packet. Router memory is oversubscribed. Reduce the number of VLANs or increase the hello delay and/or the override interval to reduce the number of simultaneous packet transmissions. If the number of flows exceeds 2046, the excess flows are routed in software, which reduces the number of packet transmissions. In this case, reducing the number of flows by moving some clients to other routers can help.</td>
</tr>
<tr>
<td>- Failed to initialize text-str as a call back routine (counter)</td>
<td>Indicates an internal error. Report the incident to your customer care center and reinstall the router software.</td>
</tr>
<tr>
<td>- I/F configured with IP ip-address on vid vlan-id (counter)</td>
<td>Indicates that the interface (VLAN) has been configured with the indicated IP address. At boot-up or when an IP address is changed, the switch generates this message for each PIM-configured VLAN.</td>
</tr>
<tr>
<td>- I/F removal with IP ip-addr on vid vlan-id (counter)</td>
<td>Indicates that a PIM interface (VLAN) has been removed from the router as a result of an IP address change or removal.</td>
</tr>
<tr>
<td>Message</td>
<td>Meaning</td>
</tr>
<tr>
<td>------------------------------------------------------------------------</td>
<td>-----------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
</tbody>
</table>
| • MCAST flow multicast-addresssource-address not rteing (rsc low) (counter) | The indicated multicast flow is not routing. The routing switch is low on memory resources as a result of too many flows for the number of configured VLANs. Remedies include one or more of the following:  
  • Reduce the number of configured VLANs by moving some VLANs to another router.  
  • Free up system resources by disabling another feature, such as one of the spanning-tree protocols or either the RIP or the OSPF routing protocol. (Unless you are using static routes, you will need to retain a minimum of one unicast routing protocol.) Another option that may help is to reduce the number of configured QoS filters.  
  • Move some hosts that create multicast demand to another router. |
| • MCAST MAC add for mac-address failed (counter)                       | • Indicates a hardware problem. Check the cabling and router ports.                                                                      |
| • Multicast Hardware Failed to Initialize (counter)                    | • Indicates a hardware failure that halts hardware processing of PIM traffic. The software will continue to process PIM traffic at a slower rate. Contact your customer care center. |
| • No IP address configured on VID vlan-id (dup-msg-cnt)                | • PIM has detected a VLAN without an IP address. Configure an IP address on the indicated VLAN.                                           |
| • Pkt dropped from ip-address , (cause ) vid vlan-id (counter)         | A PIM packet from ip-address was dropped because of one of the following causes:  
  • No PIM interface on the VLAN  
  • Bad packet length  
  • Bad IP header length  
  • Bad IP total length |
<p>| • Pkt rcvd with a checksum error from ip-addr (counter)                | • A packet having a checksum error was received from ip-addr. Check the cabling and ports on the local and the remote routers.            |
| • Rcvd incorrect hello from ip-addr (counter)                          | • Indicates receipt of a malformed hello packet. (That is, the packet does not match the current specification.) Ensure that compatible versions of PIM-DM are being used. |
| • Rcvd text-str pkt with bad len from ip-addr (counter)                | • A peer router may be sending incorrectly formatted PIM packets.                                                                      |
| • Rcvd hello from ip-address on vid vlan-id (counter)                  | • Indicates a misconfiguration where two routers are directly connected with different subnets on the same connected interface.          |
| • Rcvd pkt from rtr ip-address, unkwn pkt type value (counter)         | • A packet received from the router at ip-address is an unknown PIM packet type. (The value variable is the numeric value received in the packet.) |
| • Rcvd pkt ver# ver-num, from ip-address, expected ver-num (counter)   | • The versions of PIM-DM on the sending and receiving routers do not match. Differing versions are typically compatible, but features not supported in both versions will not be available. |</p>
<table>
<thead>
<tr>
<th>Message</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rcvd unkwn addr fmly addr-type in text-str pkt from ip-addr (counter)</td>
<td>The router received a PIM packet with an unrecognized encoding. As of February 2004, the router recognizes IPv4 encoding.</td>
</tr>
<tr>
<td>Rcvd unkwn opt opt-nbr in text-string pkt from ip-addr (counter)</td>
<td>The router received a PIM packet carrying an unknown PIM option. The packet may have been generated by a newer version of PIM-DM or is corrupt. In most cases, normal PIM-DM operation will continue.</td>
</tr>
<tr>
<td>Send error( failure-type ) on packet-type pkt on VID vid ( counter)</td>
<td>Indicates a send error on a packet. This can occur if a VLAN went down right after the packet was sent. The message indicates the failure type, the packet type, and the VLAN ID on which the packet was sent.</td>
</tr>
</tbody>
</table>
| Unable to alloc text-str table (counter) | The router was not able to create some tables PIM-DM uses. Indicates that the router is low on memory resources. Remedies include one or more of the following:  
  - Reduce the number of configured VLANs by moving some VLANs to another router.  
  - Free up system resources by disabling another feature, such as one of the spanning-tree protocols or either the RIP or the OSPF routing protocol. (Unless you are using static routes, you will need to retain a minimum of one unicast routing protocol.) Another option that may help is to reduce the number of configured QoS filters.  
  - Move some hosts that create multicast demand to another router. |
| Unable to alloc a buf of size bytes for data-flow (counter) | Multicast routing is unable to acquire memory for a flow. Router memory is oversubscribed. Reduce the number of VLANs or the number of features in use. Remedies include one or more of the following:  
  - Reduce the number of configured VLANs by moving some VLANs to another router.  
  - Free up system resources by disabling another feature, such as one of the spanning-tree protocols or either the RIP or the OSPF routing protocol. (Unless you are using static routes, you will need to retain a minimum of one unicast routing protocol.) Another option that may help is to reduce the number of configured QoS filters.  
  - Move some hosts that create multicast demand to another router. |
| Unable to alloc a msg buffer for text-message (counter) | Multicast routing is unable to acquire memory for a flow. Router memory is oversubscribed. Reduce the number of VLANs or the number of features in use. Remedies include one or more of the following:  
  - Reduce the number of configured VLANs by moving some VLANs to another router.  
  - Free up system resources by disabling another feature, such as one of the spanning-tree protocols or either the RIP or the OSPF routing protocol. (Unless you are using static routes, you will need to retain a minimum of one unicast routing protocol.) Another option that may help is to reduce the number of configured QoS filters.  
  - Move some hosts that create multicast demand to another router. |

**Applicable RFCs**

[68]
PIM is compatible with these RFCs:

- RFC 3376 - Internet Group Management Protocol, Version 3
- RFC 2365 - Administratively Scoped IP Multicast
- RFC 2932 - Multicast Routing MIB, with exceptions, see "Exceptions to Support for RFC 2932 - Multicast Routing MIB".
- RFC 2933 - IGMP MIB
- RFC 2934 - Protocol Independent Multicast MIB for IPv4

**Exceptions to Support for RFC 2932 - Multicast Routing MIB**

These MIB objects are not supported:

- ipMRoutInterfaceRateLimit
- ipMRoutInterfaceInMcastOctets
- ipMRoutInterfaceOutMcastOctets
- ipMRoutInterfaceHCInMcastOctets
- ipMRoutInterfaceHCOutMcastOctets
- ipMRoutBoundaryTable
- ipMRoutBoundaryEntry
- ipMRoutBoundaryIfIndex
- ipMRoutBoundaryAddress
- ipMRoutBoundaryAddressMask
- ipMRoutBoundaryStatus OBJECT-TYPE
- ipMRoutScopeNameTable
- ipMRoutScopeNameEntry
- ipMRoutScopeNameAddress
- ipMRoutScopeNameAddressMask
- ipMRoutScopeNameLanguage
- ipMRoutScopeNameString
- ipMRoutScopeNameDefault
- ipMRoutScopeNameStatus
For introductory information, see PIM-SM overview.
For information on PIMv6-SM, see the IPv6 Guide for your switch.

Configuring router protocol independent multicast (PIM)

For more information, see Configuration steps for PIM-SM.

The following steps configure PIM-SM in the router PIM context (switch(pim) #):

Procedure

1. Specify the VLAN interface to advertise as the Bootstrap Router (BSR) candidate and enable the router to advertise itself as a candidate BSR in a PIM-SM domain. (Use the command `bsr-candidate source-ip-vlan [vid]`)

2. **Optional:** To make BSR candidate selection occur quickly and predictably, set a different priority on each BSR candidate in the domain. (Use the command `bsr-candidate priority`.)

3. Do one of the following to configure RP operation:

   - **Recommended:** Enable Candidate Rendezvous Point (C-RP) operation and configure the router to advertise itself as a C-RP to the BSR for the current domain. This step includes the option to allow the C-RP to be a candidate for either all possible multicast groups or for up to four multicast groups and/or ranges of groups. Use the command `rp-candidate source-ip-vlan [vid] [group-addr/group-mask]`

   - **Optional:** Use the command `rp-address [ip-addr] [group-addr/group-mask]` to statically configure the router as the RP for a specified multicast group or range of multicast groups. (This must be configured on all PIM-SM routers in the domain.)

4. **Optional:** In the PIM router context, change one or more of the traffic control settings in the following table.

<table>
<thead>
<tr>
<th>Options accessed in router PIM context</th>
<th>Operation</th>
</tr>
</thead>
<tbody>
<tr>
<td>rp-candidate group-prefix [group-addr/group-mask]</td>
<td>Enter an address and mask to define an additional multicast group or a range of groups.</td>
</tr>
<tr>
<td>rp-candidate hold-time [30-255]</td>
<td>Tells the BSR how long it should expect the sending C-RP router to be operative. Default: 150; 0 if router is not a candidate</td>
</tr>
<tr>
<td>rp-candidate priority [0-255]</td>
<td>Changes the priority for the C-RP router. When multiple C-RPs are configured for the same multicast groups, the priority determines which router becomes the RP for such groups. A smaller value means a higher priority. Default: 192</td>
</tr>
</tbody>
</table>
Options accessed in router PIM context | Operation
---|---
no spt-threshold  
**Changing the shortest-path tree (SPT) operation** | Disable or enable the router's ability to switch multicast traffic flows to the shortest path tree. Default: enabled

join-prune-interval [5-65535]  
**Changing the interval for PIM-SM neighbor notification** | Option: Globally change the interval for the frequency at which join and prune messages are forwarded on the router's VLAN interfaces. Default: 60 seconds

trap [neighbor-loss | hardware-mrt-full | software-mrt-full | all] | Option: Enable or disable PIM traps. Default: disabled

### Configuring PIM-SM on the router

#### Global configuration context for supporting PIM-SM

Before configuring specific PIM-SM settings, it is necessary to enable IP routing, IP multicast routing, an IP routing protocol, and PIM in the global configuration context. Also, if the router operates as an edge router for any end points (receivers) expected to join multicast groups, it is also necessary to enable IGMP on the VLANs supporting such receivers.

#### Configuring global context commands

**Note:** PIM-SM operation requires an IP routing protocol enabled on the router. You can use RIP, OSPF, and/or static routing. The examples in this section use RIP.

**Syntax:**

```plaintext
ip routing  
no ip routing
```

Enables IP routing on the router.

The `no` form of the command disables IP routing. Note that before disabling IP routing, it is necessary to disable all other IP routing protocols on the router.

( Default: Disabled)

**Syntax:**

```plaintext
ip multicast-routing  
no ip multicast-routing
```

Enables or disables IP multicast routing on the router. IP routing must first be enabled. Note that router PIM must be disabled before disabling IP multicast routing.

( Default: Disabled)

**Syntax:**

```plaintext
router [ospf | rip]  
no router [ospf | rip]  
ip route [ip-addr/mask-len] [ip-addr | vlan | reject | null-route]  
no ip route [ip-addr/mask-len] [ip-addr | vlan | reject | null-route]
```

These commands are the options for the IP routing protocol required to support PIM operation.
Syntax:

```
no router pim [[enable] | [disable]]
```

Puts the CLI into the PIM context level. IP routing must be enabled before enabling PIM.
The `no router pim` command deletes the PIM configuration. (Default: Disabled)

- `[enable]` Enables PIM globally.
- `[disable]` Disables PIM globally. Disabling PIM does not delete the PIM configuration.

**Configuring for PIM support at the global level**

Using the topology shown in the following figure, router "B" is directly connected to the DR for multicast group "X." In this case, suppose that you want to globally configure router "B" for PIM operation. On the global level, you would enable the following:

- IP routing
- IP multicast routing
- An IP routing protocol (RIP, OSPF, or static routing; use RIP for this example)

**PIM-SM domain with SPT active to support a host that has joined a multicast group**

![Diagram of PIM-SM domain with SPT active]

**Global configuration for supporting PIM-SM operation**

```
switch(config)# ip routing
switch(config)# ip multicast-routing
switch(config)# router rip
switch(rip)# exit
switch(config)# router pim
switch(pim)# exit
switch(config)#
```

**Displaying the running configuration**
VLAN context commands for configuring PIM-SM

PIM-SM must be configured on at least one VLAN in the router before it can be configured as a C-BSR or a C-RP.

Enabling or disabling IGMP in a VLAN

IGMP must be enabled in VLANs on edge routers where multicast receivers (end points) are connected and will be requesting to join multicast groups.

Syntax:

no ip igmp
no vlan [vid] ip igmp

Enables or disables IGMP operation in the current VLAN. Configuring IGMP on the router is required in VLANs supporting edge router operation.

Enabling or disabling PIM-SM per-VLAN

Syntax:

ip pim-sparse [ip-addr[any | ip-addr ]]
vlan [vid] ip pim-sparse [ip-addr[any | ip-addr ]]
no [vlan [ vid ]] ip pim-sparse

This command enables or disables PIM-SM in the designated VLAN interface and sets the source (and designated router) IP address for PIM-SM packets sent from the interface. Executing the command without specifying an IP address option causes the router to default to the any option, below. (Default: PIM-SM disabled)

ip-addr any

Enables the router to dynamically determine from the VLAN’s current IP configuration the source IP address to use for PIM-SM packets sent from the VLAN interface.

Using this command after a source IP address has already been set does not change that setting.
Specifies one of the VLAN’s currently existing IP addresses for use as the source IP address for PIM-SM packets sent from the VLAN interface.

Note that ip-addr must first be statically configured on the VLAN.

To change an existing source IP address setting, you must use this command option.

### Changing the interval for PIM-SM neighbor notification

**Syntax:**

```plaintext
ip pim-sparse hello-interval [5-300]
vlan vid ip pim-sparse hello-interval [5-300]
```

Changes the frequency at which the router transmits PIM hello messages on the current VLAN. The router uses hello packets to inform neighbor routers of its presence.

The router also uses this setting to compute the **Hello hold time**, which is included in hello packets sent to neighbor routers. **Hello hold time** tells neighbor routers how long to wait for the next hello packet from the router. If another packet does not arrive within that time, the router removes the neighbor adjacency on that VLAN from the routing table, which removes any flows running on that interface.

Shortening the hello interval reduces the **Hello hold time**. This changes how quickly other routers will stop sending traffic to the router if they do not receive a new hello packet when expected. For example, if multiple routers are connected to the same VLAN and the router requests multicast traffic, all routers on the VLAN receive that traffic. (Those that have pruned the traffic will drop it when they receive it.) If the upstream router loses contact with the router receiving the multicast traffic (that is, fails to receive a hello packet when expected), the shorter hello interval causes it to stop transmitting multicast traffic onto the VLAN sooner, resulting in less unnecessary bandwidth use.

(Defualt: 30 seconds)

### Changing the randomized delay setting for PIM-SM neighbor notification

**Syntax:**

```plaintext
ip pim-sparse hello-delay [0-5]
vlan [vid] ip pim-sparse hello-delay [0-5]
```

Changes the maximum time in seconds before the router actually transmits the initial PIM hello message on the current VLAN. In cases where a new VLAN activates with connections to multiple routers, if all of the connected routers sent hello packets at the same time, the receiving router could become momentarily overloaded. This value randomizes the transmission delay to a time between 0 and the **hello delay** setting. Using 0 means no delay.

After the router sends the initial hello packet to a newly detected VLAN interface, it sends subsequent hello packets according to the current **Hello Interval** setting.

Not used with the no form of the ip pim command.

(Defualt: 5 seconds)

### Enabling or disabling lan prune delay

**Syntax:**

```plaintext
ip pim-sparse lan-prune-delay
no ip pim-sparse lan-prune-delay
vlan [vid] ip pim-sparse lan-prune-delay
no vlan [vid] ip pim-sparse lan-prune-delay
```

Enables the LAN prune delay option on the current VLAN. With lan-prune-delay enabled, the router informs downstream neighbors how long it will wait before pruning a flow after receiving a prune request.
Other downstream routers on the same VLAN must send a join to override the prune before the lan-prune-delay time if they want the flow to continue. This prompts any downstream neighbors with multicast receivers continuing to belong to the flow to reply with a join. If no joins are received after the lan-prune-delay period, the router prunes the flow.

The propagation-delay and override-interval settings (below) determine the lan-prune-delay setting. Uses the no form of the command to disable the LAN prune delay option. (Default: Enabled)

**Changing the Lan-prune-delay interval**

**Syntax:**

```plaintext
ip pim-sparse propagation-delay [250-2000]
ip pim-sparse override-interval [500-6000]
vlan [vid] ip pim-sparse override-interval [500-6000]
```

A router sharing a VLAN with other multicast routers uses these two values to compute the lan-prune-delay setting (above) for how long to wait for a PIM-SM join after receiving a prune packet from downstream for a particular multicast group.

**Multiple routers sharing VLAN**

A network may have multiple routers sharing VLAN "X." When an upstream router is forwarding traffic from multicast group "X" to VLAN "Y," if one of the routers on VLAN "Y" does not want this traffic, it issues a prune response to the upstream neighbor. The upstream neighbor then goes into a prune pending state for group "X" on VLAN "Y." (During this period, the upstream neighbor continues to forward the traffic.) During the pending period, another router on VLAN "Y" can send a group "X" join to the upstream neighbor. If this happens, the upstream neighbor drops the prune pending state and continues forwarding the traffic. But if no routers on the VLAN send a join, the upstream router prunes group "X" from VLAN "Y" when the lan-prune-delay timer expires.

(Defaults: propagation-delay = 500 milliseconds; override-interval = 2500 milliseconds)

**Neighbor timeout**

**Syntax:**

```plaintext
ip pim-sparse nbr-timeout [60-65536]
```

**Changing the DR priority**

**Syntax:**

```plaintext
ip pim-sparse dr-priority [0-4294967295]
```

This command changes the router priority for the DR election process in the current VLAN. A numerically higher value means a higher priority. If the highest priority is shared by multiple routers in the same VLAN, the router with the highest IP address is selected as the DR.

A 0 (zero) value disables DR operation for the router on the current VLAN.

(Range: 0 - 2147483647; Default: 1)

**Configuring PIM-SM support in a VLAN context**

PIM-SM support must be configured in each VLAN where you want PIM-SM forwarding of multicast traffic. This illustrates the following per-VLAN configuration steps:
Enabling PIM-SM on VLAN 120 and allowing the default any option to select a source IP address for PIM-SM packets forwarded from this VLAN. (Because the VLAN in this example is configured with only one IP address—120-10.10.2—it is this address that will be used for the source.)

- Increasing the DR priority on this VLAN from the default 1 to 100.
- Leaving the other per-VLAN PIM-SM fields in their default settings.

**Example of Enabling PIM-SM in a VLAN**

```
Switch(config)# vlan 120
Switch(vlan-120)# ip pim sparse
Switch(vlan-120-pim-sparse)# dr-priority 100
Switch(vlan-120-pim-sparse)# show ip pim interface 120
```

**Router PIM context commands for configuring PIM-SM operation**

This section describes the commands used in the Router PIM context to:

- Enable or disable SNMP trap status for PIM events (default: disabled)
- Configure candidate BSR operation
- Configure C-RP operation or the (optional) static RP operation

**NOTE**

Before configuring BSR, RP, and SNMP trap operation for PIM-SM, it is necessary to enable PIM-SM on at least one VLAN on the router.

**Configuring a BSR candidate**

Selecting the VLAN interface to advertise as a BSR candidate.

**Syntax:**

```
bsr-candidate source-ip-vlan [vid]
no bsr-candidate source-ip-vlan [vid]
router pim bsr-candidate source-ip-vlan [vid]
no router pim bsr-candidate source-ip-vlan [vid]
```

Configures the router to advertise itself as a candidate PIM-SM BSR on the VLAN interface specified by source-ip-vlan [vid], and enables BSR candidate operation. This makes the router eligible to be elected as the BSR for the PIM-SM domain in which it operates. Note that one BSR candidate VLAN interface is allowed per-router. The no form of the command deletes the BSR source IP VLAN configuration and also disables the router from being a BSR candidate, if this option has been enabled. (See the BSR-candidate command, below.)

**Enabling or disabling a BSR Candidate**

Enable or disable BSR candidate operation on a router.
**Syntax:**

bsr-candidate
no bsr-candidate
router pim bsr-candidate
no router pim bsr-candidate

Disables or re-enables the router for advertising itself as a Candidate-BSR on the VLAN interface specified by source-ip-vlan [vid]. This command is used to disable and re-enable BSR candidate operation after the bsr-candidate source-ip-vlan [vid] command has been used to enable C-BSR operation on the router. (This command operates only after the BSR source-ip-VLAN ID has been configured.)

(Default: Disabled)

**Changing the priority setting**

Changing the priority setting for a BSR-candidate router.

**Syntax:**

bsr-candidate priority [0-255]
router pim bsr-candidate priority [0-255]
no router pim bsr-candidate priority [0-255]

Specifies the priority to apply to the router when a BSR election process occurs in the PIM-SM domain. The candidate with the highest priority becomes the BSR for the domain. If the highest priority is shared by multiple routers, the candidate having the highest IP address becomes the domain's BSR. Zero (0) is the lowest priority. To make BSR selection easily predictable, use this command to assign a different priority to each candidate BSR in the PIM-SM domain.

(Default: 0; Range 0-255)

---

_Disabling PIM-SM on the elected BSR or disabling the C-BSR functionality on the elected BSR causes the router to send a Bootstrap Message (BSM) with a priority setting of 0 to trigger a new BSR election. If all BSRs in the domain are set to the default priority (0), the election will fail because the result is to re-elect the BSR that has become unavailable. For this reason, it is recommended that all C-BSRs in the domain be configured with a bsr-candidate priority greater than 0._

---

**Changing the distribution**

Changing the distribution of multicast groups across a domain.

**Syntax:**

bsr-candidate hash-mask-length [1-32]
no router pim bsr-candidate hash-mask-length [1-32]

Controls distribution of multicast groups among the C-RPs in a domain where there is overlapping coverage of the groups among the RPs. This value specifies the length (number of significant bits) taken into account when allocating this distribution. A longer hash-mask-length results in fewer multicast groups in each block of group addresses assigned to the various RPs. Because multiple blocks of addresses are typically assigned to each C-RP, this results in a wider dispersal of addresses and enhances load-sharing of the multicast traffic of different groups being used in the domain at the same time.

(Default: 30; Range 1-32)

**Changing the message interval**

Changing the BSR message interval.

**Syntax:**

bsr-candidate bsm-interval [5-300]
no bsr-candidate bsm-interval [5-300]

Specifies the interval in seconds for sending periodic RP-Set messages on all PIM-SM interfaces on a router operating as the elected BSR in a domain.

<table>
<thead>
<tr>
<th>NOTE</th>
</tr>
</thead>
<tbody>
<tr>
<td>This setting must be smaller than the rp-candidate hold-time settings (range of 30 to 255; default 150) configured in the RPs operating in the domain.</td>
</tr>
</tbody>
</table>

( Default: 60; Range 5-300 )

**Configuring C-RPs on PIM-SM routers**

An RP candidate advertises its availability, IP address, and the multicast group or range of groups it supports. The commands in this section are used to configure C-RP operation. The sequence of steps is as follows:

1. Specify the source IP VLAN.
2. Enable C-RP operation.
3. **Option:** enable or disable specific multicast address groups.

<table>
<thead>
<tr>
<th>NOTE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Before configuring BSR, RP, and SNMP trap operation for PIM-SM, it is necessary to enable PIM-SM on at least one VLAN on the router.</td>
</tr>
</tbody>
</table>

**Specifying the source IP VLAN (optionally configuring multicast groups or range of groups)**

Specifying the source IP VLAN ID automatically configures the C-RP to support all multicast groups (unless you include an individual group or range of groups in the command.) The recommended approach is to allow all multicast groups unless you have a reason to limit the permitted groups to a specific set.

**Syntax:**

```
no rp-candidate source-ip-vlan [vid] [group-prefix group-addr/mask]
no router pim rp-candidate source-ip-vlan [vid] [group-prefix group-addr/mask]
```

These commands configure C-RP operation in the following way:

- Specify the VLAN interface from which the RP IP address will be selected for advertising the router as an RP candidate.

<table>
<thead>
<tr>
<th>NOTE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Only one VLAN on the router can be configured for this purpose at any time.</td>
</tr>
</tbody>
</table>

- Enable the router as an RP candidate.
- Specify the multicast groups for which the router is a CRP. (Default: Disabled.)

<table>
<thead>
<tr>
<th>NOTE</th>
</tr>
</thead>
<tbody>
<tr>
<td>When executed without specifying a multicast group or range of groups, the resulting RP candidate defaults to allow support for all valid multicast groups.</td>
</tr>
</tbody>
</table>

Additionally, the following commands may be required:

- To later add to or change multicast groups, or to delete multicast groups, use the command `rp-candidate group-prefix [group-addr | group-mask].`
- To disable C-RP operation without removing the current CRP configuration, use the command `no rp-`
candidate.

- The **no** form of these commands:
  - Deletes the RP source IP VLAN configuration.
  - Deletes the multicast group assignments configured on the router for this RP.
  - Disables the router from being an RP candidate.

The `<vid>` command identifies the VLAN source of the IP address to advertise as the RP candidate address for the router.

The command `group-prefix [group-addr/mask]` specifies the multicast group(s) to advertise as supported by the RP candidate. Use this option when you want to enable the C-RP and simultaneously configure it to support a subset of multicast addresses or ranges of addresses instead of all possible multicast addresses.

A group prefix can specify all multicast groups (224.0.0.0 to 239.255.255.255), a range (subset) of groups, or a single group. A given address is defined by its nonzero octets and mask. The mask is applied from the high end (leftmost) bits of the address and must extend to the last nonzero bit in the lowest-order, nonzero octet. Any intervening zero or nonzero octet requires eight mask bits. Following are examples.

**228.0.0.64/26:**
Defines a multicast address range of 228.0.0.64 through 228.0.0.127. (The last six bits of the rightmost octet are wildcards.)

**228.0.0.64/30:**
Defines a multicast address range of 228.0.0.64 through 228.0.0.67. (The last two bits of the rightmost octet are wildcards.)

**228.0.0.64/32:**
Defines a single multicast address of 228.0.0.64. (There are no wildcards in this group prefix.)

**228.0.0.64/25:**
Creates an error condition caused by the mask failing to include the last (rightmost) nonzero bit in the lowest-order, nonzero octet. (That is, this mask supports an address of 228.0.0.128, but not 228.0.0.64.)

---

The larger the mask, the smaller the range of multicast addresses supported. A mask of 32 bits always specifies a single multicast address. For example 230.0.15.240/32 defines a single multicast address of 230.0.15.240.

---

**Enabling or disabling C-RP operation**

Use this command when the router is already configured with a source IP VLAN ID and you want to enable or disable C-RP operation on the router.

**Syntax:**

```
no rp-candidate
```

Enables C-RP operation on the router. Requires that the source IP VLAN is currently configured, but disabled.

The **no** form of the command disables the currently configured C-RP operation, but does not change the configured C-RP settings.

**Adding or deleting a multicast group address**

Use this command if you need to modify the multicast address group configuration for a C-RP on the router.
Syntax:
no rp-candidate group-prefix [group-addr | group-mask]

Adds a multicast group address to the current C-RP configuration. Requires that the source IP VLAN (See Specifying the source IP VLAN (optionally configuring multicast groups or range of groups)) is already configured. The no form of the command removes a multicast group address from the current C-RP configuration. This command does not enable or disable RP candidate operation.

An RP candidate supports up to four separate multicast address groups. If only one group-prefix address exists in the router PIM configuration, you cannot delete it unless you first add another group-prefix address.

Changing the C-RP hold-time

Hold-time is included in the advertisements the C-RP periodically sends to the domain’s elected BSR, and updates the BSR on how long to wait after the last advertisement from the reporting RP before assuming that it has become unavailable.

Syntax:
rp-candidate hold-time [30-255]

Changes the hold time a C-RP includes in its advertisements to the BSR. Also, if C-RP is configured, but disabled, this command re-enables it.

(Default: 150 seconds; Range: 30-255 seconds.)

Changing a C-RP's election priority

This priority is significant when multiple C-RPs in a given domain are configured to support one or more of the same multicast groups.

Syntax:
rp-candidate priority [0-255]

Changes the current priority setting for a C-RP. Where multiple C-RPs are configured to support the same multicast group(s), the candidate having the highest priority is elected. Zero (0) is the highest priority, and 255 is the lowest priority.

(Default: 192)

Enabling, disabling, or changing router PIM notification traps

Syntax:
no router pim trap [all | neighbor-loss | hardware-mrt-full | software-mrt-full]

Enables and disables the following PIM SNMP traps:
all

Enable/Disable all PIM notification traps.
(Default: Disabled)
neighbor-loss

Enable/Disable the notification trap sent when the timer for a multicast router neighbor expires and the switch has no other multicast router neighbors on the same VLAN with a lower IP address.
(Default: Disabled)
hardware-mrt-full
Enable/Disable notification trap sent when the hardware multicast routing table (MRT) is full (2046 active flows.) In this state, any additional flows are handled by the software MRT, which increases processing time for the affected flows.
(Default: Disabled)

**software-mrt-full**

Enable/Disable notification trap sent when the router's software MRT is full (that is, when routing resources for active flows are exhausted.) Note that in this state, the router does not accept any additional flows.
(Default: Disabled)

---

**Trap operation requires configuring an SNMP trap receiver by using the**

```
    snmp-server host [ip-addr]
```

command at the global configuration level.

---

**Changing the global join-prune interval on the router**

**Syntax:**

```
    router pim join-prune-interval [5-65535]
```

Sets the interval in seconds at which periodic PIM-SM join/prune messages are to be sent on the router's PIM-SM interfaces. This setting is applied to every PIM-SM interface on the router.
(Default: 60 seconds)

---

All routers in a PIM-SM domain should have the same join-prune-interval setting.

---

**Changing the shortest-path tree (SPT) operation**

Generally, using the SPT option eliminates unnecessary levels of PIM-SM traffic in a domain. However, in cases where it is necessary to tightly control the paths used by PIM-SM flows to edge switches, disabling SPT maintains the flows through their original C-RPs regardless of whether shorter paths exist.

**Syntax:**

```
    router pim spt-threshold
    no router pim spt-threshold
```

When the router is the edge router for a receiver requesting to join a particular multicast group, this command enables or disables the capability of the router to convert the group's traffic from the RPT to the SPT.

See [Restricting multicast traffic to RPTs](#).
(Default: Enabled)

---

**Statically configuring an RP to accept multicast traffic**

A given static RP entry should be manually configured on all routers in the PIM-SM domain.

**Syntax:**

```
    router pim rp-address [rp-ip-addr] [group-addr/group-mask] [override]
    no router pim rp-address [rp-ip-addr] [group-addr/group-mask] [override]
    [rp-ip-addr]
```
Statically specifies the IP address of the interface to use as an RP. Up to eight static RP IP addresses can be configured. (Each address can be entered multiple times for different multicast groups or group ranges.)

[group-addr/group-mask]

Specifies the multicast group or range of contiguous groups supported by the statically configured RP. Up to eight multicast group ranges can be configured.

[override]

Where a static RP and a C-RP are configured to support the same multicast group(s) and the multicast group mask for the static RP is equal to or greater than the same mask for the applicable C-RPs, this command assigns the higher precedence to the static RP, resulting in the C-RP operating only as a backup RP for the configured group. Without override, the C-RP has precedence over a static RP configured for the same multicast group(s.)

Configuring PIM-SM support in the router PIM context

This example assumes the following:

- IP routing, IP multicast routing, and at least one routing method (RIP, OSPF, and/or static IP routes) are already configured in the global configuration context.
- An IP routing method (RIP or OSPF) and PIM-sparse are already configured in the static VLAN context on which you want to support PIM-SM operation.

Routers configured for C-RP operation can also be configured for C-BSR operation.

Use of static RP operation must be identically configured on all PIM-SM routers in the domain.

Example of enabling PIM-SM in the router PIM context illustrates the following configuration steps for the router PIM context:

- Enabling BSR operation on the router, including specifying a source IP address.
- Enabling C-RP operation on the router.
- Replacing the default multicast group range (all) with a smaller range (231.128.24.0/18) and a single group address (230.255.1.1/32.)
- Enabling static RP with override on this router for a single group address (231.128.64.255/32) within the range of the C-RP support for the 231.128.24.0 group.
- Leaving the other router PIM fields in their default settings.

Figure 5 Example of enabling PIM-SM in the router PIM context
The next figure illustrates the results of the above commands in the router's running configuration.

**Configuration results of the commands in Example of enabling PIM-SM in the router PIM context**

```
switch(pim)# show running configuration:
router pim
  bsr-candidate
  bsr-candidate source-ip-vlan 120
  bsr-candidate priority 1
  rp-address 120.10.10.1 231.128.64.0/18 255.255.255.255
  rp-candidate
  rp-candidate source-ip-vlan 120
  rp-candidate group-prefix 230.255.1.1 255.255.255.255
  rp-candidate group-prefix 231.128.64.0 255.255.192.0
  rp-candidate hold-time 150
exit
```

**PIM RPF override configuration**

**Overview of PIM RPF**

Reverse Path Forward (RPF) checking is a core multicast routing mechanism which ensures that the multicast traffic received has arrived on the expected router interface derived from the L3 table prior to further processing. If the RPF check fails for a multicast packet, the packet is discarded.

For traffic arriving on the SPT, the expected incoming interface for a given source/group multicast flow is the interface towards the source address of the traffic (as determined by the unicast routing system.) For traffic arriving on the RP tree, the expected incoming interface is the interface towards the RP.

RPF override is a feature that allows the override of the normal RPF lookup mechanism and indicates to the router that it may accept multicast traffic on an interface, other than the normally selected interface by the RPF lookup mechanism. This includes accepting traffic from a source directly connected to the router when the source IP address is invalid for the subnet or VLAN to which it is connected. Traffic may also be accepted from a valid PIM neighbor that is not on the reverse path towards the source of the received multicast traffic.

RPF checking is applied to all multicast traffic and is significant in preventing network loops. Up to eight manual RPF overrides can be specified.

---

*Note: The static RP takes precedence over the C-RP for multicast groups in the range of 231.128.64.0/18 because the mask configured for the static RP meets the criteria of being either equal to or greater than the mask configured for the same group in the C-RP. For example, if the mask for the static-RP was 17 or less, the override would not take effect (even though configured), and the C-RP configuration would take precedence.*
The manually configured static multicast RPF override is restored on subsequent reboots. The command is executed in PIM context.

**rpf-override**

```plaintext
no rpf-override [source-ip/addr/mask-length] [rpf-ip-addr]
```

Add, edit, or delete up to eight RPF override entries. The multicast RPF override has a multicast source address [source-ip/addr/mask-length] and an RPF address [rpf-ip-addr] pair.

The no form of the command deletes the RPF override.

**NOTE**

Only host-specific addresses are supported (i.e. “/32” addresses.)

[source-ip/addr]

The IPv4 address of the host from which the multicast flow originated.

[mask-length]

The length, in bits, of the mask used to indicate the range of addresses from [source-ip/addr] to which the RPF override command applies. Currently, only a 32-bit mask is supported, that is, only one host per entry. Eight individual entries are supported.

[rpf-ip-addr]

The IPv4 address indicating one of two distinct RPF candidates:

1. A valid PIM neighbor address from which forwarded multicast traffic is accepted with a source address of [source-ip/addr].
2. A local router address on a PIM-enabled VLAN to which [source-ip/addr] is directly connected. The local router will assume the role of DR for this flow and registers the flow with an RP, if configured.

The following example shows how to configure a manual multicast RPF override and saving it in a config file.

```plaintext
switch(config)# ip routing
switch(config)# ip multicast-routing
switch(config)# router pim
switch(pim)# rpf-override 10.1.1.1/32 11.2.2.1
switch(pim)# write mem
```

**Displaying configured RPF overrides**

You can display the configured RPF overrides with the `show` command.

**Syntax:**

```plaintext
show ip pim rpf-override [source | source ip-address]
```

Displays the configured RPF override entries.

[source ip-address]

Displays the RPF overrides for a specific IP address. This can be useful when troubleshooting potential RPF misconfigurations.

**Displaying the configured RPF overrides**

```plaintext
switch(config)# show ip pim rpf-override
Static RPF Override
```
### Specifying the source parameter to troubleshoot misconfigurations

```plaintext
switch(pim)# show ip pim rpf-override source 10.1.1.1
Static RPF Override
  Multicast Source  RPF IP Address
  10.1.1.1/32       11.2.2.1
```

### Displaying PIM route data

**Syntax:**

```plaintext
show ip [mroute]
```

Without parameters, lists multicast route entries in the following situations:

- When the PIM-DM router is actively forwarding a multicast flow out an interface (VLAN.)
- On a PIM-DM originator router (source directly connected) when traffic is entering the router but not forwarding

**NOTE**

The neighbor field will be empty in this case.

- On a PIM-DM Non-originator router for a short duration after a flow’s initial flood/prune cycle is seen. This entry is cleared after 5 minutes unless the flow is connected within that time period.

**[Group Address]**

The multicast group IP address for the specific flow (source-group pair.)

**[Source Address]**

The unicast address of the flow’s source.

**[neighbor]**

The IP address of the upstream multicast router interface (VLAN) from which the multicast flow is coming. A blank field indicates that the multicast source is directly connected to the router.

**[VLAN]**

The interface on which the router receives the multicast flow.

### Showing the route entry data on the “#2” routing switch

The next figure displays the `show ip mroute` output on the “#2” routing switch shown in Example of configuring PIM-DM operation at the VLAN level. This case illustrates two multicast groups from the same multicast source.

```plaintext
switch(config)# show ip mroute
IP Multicast Route Entries
Total number of entries : 2
Group Address Source Address Neighbor VLAN
----------------------- --------------- -------------
239.255.255.1       10.27.30.2    10.29.30.1 29
239.255.255.5       10.27.30.2    10.29.30.1 29
```

### Listing basic route data for active multicast groups

**Syntax:**

```plaintext
```
show ip mroute

Lists the following data for all VLANs actively forwarding multicast traffic, or for VLANs receiving registered but non-forwarding traffic on an RP.

**Group Address**
The multicast group IP address of the specific flow (source-group pair.)

**Source Address**
The unicast address of the flow’s source.

**Neighbor**
The IP address of the upstream multicast router interface (VLAN) from which the multicast traffic is coming. A blank field for a given multicast group indicates that the multicast server is directly connected to the router.

**VLAN**
The interface on which the router received the multicast flow.

The following examples display the show ip mroute output illustrating three different cases:

### Showing source-DR PIM router
Source-DR PIM router. A flow’s Neighbor field is not empty for a PIM Router with a directly connected source.

```
switch# sh ip pim mroute
IP Multicast Route Entry
Group Address  : 226.94.2.2
Source Address  : 70.70.70.10
Neighbor        : 70.70.70.10
VLAN            : 70
Up Time (sec)   : 72
Expire Time (sec) : 292
Multicast Routing Protocol : PIM-SM
Unicast Routing Protocol : connected
```

### Showing intermediate PIM router
Flows show their adjacent PIM neighbor towards the source.

```
switch(config)# show ip mroute
IP Multicast Route Entries
Total number of entries : 2
<table>
<thead>
<tr>
<th>Group Address</th>
<th>Source Address</th>
<th>Neighbor</th>
<th>VLAN</th>
</tr>
</thead>
<tbody>
<tr>
<td>239.255.12.42</td>
<td>10.0.0.10</td>
<td>20.0.0.1</td>
<td>20</td>
</tr>
<tr>
<td>239.255.255.255</td>
<td>10.0.0.10</td>
<td>20.0.0.1</td>
<td>20</td>
</tr>
</tbody>
</table>
```

### Showing new RP special case
RP special case: When run on a RP, registered but non-forwarding flows are displayed without a neighbor value. This is identical in appearance to a direct-connected source, but on an RP this indicates the unique registered, non-forwarding condition.

```
switch(config)# show ip mroute
IP Multicast Route Entries
Total number of entries : 2
<table>
<thead>
<tr>
<th>Group Address</th>
<th>Source Address</th>
<th>Neighbor</th>
<th>VLAN</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
```

The following examples display the show ip mroute output illustrating three different cases:
Listing all VLANs having currently active PIM flows

Syntax:
show ip mroute interface [vid]

This command displays exactly the same output as the command show ip pim interface vid (See Listing currently configured PIM interfaces.)

Viewing PIM-specific data

The commands in this section display PIM-specific multicast routing information for IP multicast groups detected by the router.

Viewing the current PIM status and global configuration

Syntax:
show ip pim

Displays PIM status and global parameters.

PIM Status

Shows either Enabled or Disabled.

State Refresh Interval (sec)

Applies only to PIM-DM operation.

Join/Prune Interval

Indicates the frequency with which the router transmits join and prune messages for the multicast groups the router is forwarding.

SPT Threshold

When Enabled, indicates that, for a given receiver joining a multicast group, an edge router changes from the RPT to the SPT after receiving the first packet of a multicast flow intended for a receiver connected to the router.

When Disabled, indicates that the no spt-threshold command has been used to disable SPT operation. (See Changing the shortest-path tree (SPT) operation.)

Traps

Enables the following SNMP traps:

neighbor-loss

Sends a trap if a neighbor router is lost.

all

Enables all of the above traps.

none

No traps are set.

Output with PIM enabled

```
switch(config)# show ip pim
PIM Global Parameters
    PIM Status             : Enabled
    State Refresh Interval (sec) : 60
    Join/Prune Interval (sec) : 60
    SPT Threshold          : Enabled
```
Displaying current PIM entries existing in the multicast routing table

Syntax:
show ip pim mroute

This command displays exactly the same output as the command `show ip mroute`.

Listing currently configured PIM interfaces

Syntax:
show ip pim interface

Lists the PIM interfaces (VLANs) currently configured in the router.

VLAN
Lists the vid of each VLAN configured on the switch to support PIM-DM.

IP Address
Lists the IP addresses of the PIM interfaces (VLANs.)

Mode
Shows dense or sparse, depending on which PIM protocol is configured on the router.

Two configured PIM interfaces

switch(config)# show ip pim interface
<table>
<thead>
<tr>
<th>VLAN</th>
<th>IP Address</th>
<th>Mode</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>10.1.10.1</td>
<td>sparse</td>
</tr>
<tr>
<td>2</td>
<td>10.2.10.1</td>
<td>sparse</td>
</tr>
</tbody>
</table>

Displaying IP PIM VLAN configurations

Syntax:
show ip pim interface [vid]

Displays the current configuration for the specified VLAN (PIM interface.)

Table 10: PIM interface configuration settings

<table>
<thead>
<tr>
<th>Field</th>
<th>Default</th>
<th>Control command</th>
</tr>
</thead>
<tbody>
<tr>
<td>VLAN</td>
<td>N/A</td>
<td>vlan vid ip pim</td>
</tr>
<tr>
<td>IP</td>
<td>N/A</td>
<td>vlan vid ip pim all</td>
</tr>
<tr>
<td>Mode</td>
<td>dense</td>
<td>n/a; PIM Dense only</td>
</tr>
<tr>
<td>Hello interval (sec)</td>
<td>300</td>
<td>ip pim hello interval 5 - 30</td>
</tr>
</tbody>
</table>
**Field** | **Default** | **Control command**
--- | --- | ---
Hello delay | 5 | The router includes this value in the "Hello" packets that it sends to neighbor routers. Neighbor routers use this value to determine how long to wait for another Hello packet from the router. See Changing the interval for PIM-SM neighbor notification.
override interval (msec) | 2500 | vlan vid ip pim override-interval 500 - 6000
Propagation delay (msec) | 500 | vlan vid ip pim propagation-delay 250-2000
LAN prune delay | Yes | vlan vid ip pim lan-prune-delay
LAN delay enabled | No | Shows Yes if all multicast routers on the current VLAN interface enabled LAN-prune-delay. Otherwise, shows No.
DR priority | 1 | ip pim-sparse dr-priority 0 - 4294967295

**Showing a PIM-SM interface configured on VLAN 1**

```
switch(config)# show ip pim interface 1
PIM Interface
  VLAN : 1
  IP Address : 10.1.10.1
  Mode : sparse
    Designated Router : 10.1.10.1
  Hello Interval (sec) : 30
  Hello Delay (sec) : 5
  Override Interval (msec) : 2500 Lan Prune Delay : Yes
  Propagation Delay (msec) : 500 Lan Delay Enabled : No
  Neighbour Timeout : 180 DR Priority : 1
```

**Displaying PIM neighbor data**

These commands enable listings of either all PIM neighbors the router detects or the data for a specific PIM neighbor.

**Syntax:**

show ip pim neighbor

Lists PIM neighbor information for all PIM neighbors connected to the router:

IP Address

Lists the IP address of a neighbor multicast router.

VLAN

Lists the VLAN through which the router connects to the indicated neighbor.

Up Time
Shows the elapsed time during which the neighbor has maintained a PIM route to the router.

**Expire Time**

Indicates how long before the router ages-out the current flow (group membership.) This value decrements until:

- Reset by a state-refresh packet originating from the upstream multicast router. (The upstream multicast router issues state-refresh packets for the current group as long as it either continues to receive traffic for the current flow or receives state-refresh packets for the current flow from another upstream multicast router.
- Reset by a new flow for the current multicast group on the VLAN.
- The timer expires (reaches 0.) In this case, the switch has not received either a state-refresh packet or new traffic for the current multicast group and ages-out (drops) the group entry.

**DR Priority**

Shows the currently configured priority for DR operation on the interface.

**Listing of all PIM neighbors detected**

```
switch(config)# show ip pim neighbor

PIM Neighbors

<table>
<thead>
<tr>
<th>IP Address</th>
<th>VLAN</th>
<th>Up Time (sec)</th>
<th>Expire Time (sec)</th>
<th>DR Priority</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.10.10.2</td>
<td></td>
<td>348</td>
<td>90</td>
<td>1</td>
</tr>
<tr>
<td>10.20.10.1</td>
<td>100</td>
<td>410</td>
<td>97</td>
<td>1</td>
</tr>
</tbody>
</table>
```

**Syntax:**

```
show ip pim neighbor [ip-address]
```

Lists the same information as `show ip pim neighbor`. See

**Output for a specific PIM neighbor**

```
switch(config)# show ip pim neighbor 10.10.10.2

PIM Neighbor
IP Address : 10.10.10.2
VLAN : 100
Up Time (sec) : 678
Expire Time (sec) : 93
DR Priority : 1
```

**Display pending join requests**

Use the `show ip pim pending` command to display the pending joins on a PIM router. A pending join can be an IGMPv2 join (host join) or PIM (*,G) or (S,G) join (PIM router joins, PIM-SM only) received by a router for which there is no active multicast flow to satisfy the received join. This aids in determining what flows are being requested on the PIM network, but for which there is no data. If data availability is expected for a flow, and a join for that flow is showing as pending, this moves the troubleshooting search to the source of the flow since the routers are verified to be seeing the request for data.

**Syntax:**

```
show ip pim pending [ip-address]
```

Displays the joins received on the switch from downstream devices that want to join a specified (*,G) or (S,G) multicast group (flow) address or all multicast groups known on the switch.

A join remains in a pending state until traffic is received for the flow. The VLAN (PIM interface) on which each join was received is also displayed.

**Incoming VLAN**
ID on which a join request is received.

**Source IPv4 Address**

IP address of the source of multicast traffic in an (S,G) group.

**Incoming VLAN**

VLAN ID from which a join request is received.

**Source IPv4 Address**

IP address of the source of multicast traffic in an (S,G) group.

### Displaying BSR data

The router provides BSR information through both IP PIM and the running configuration.

### Displaying BSR status and configuration

**Syntax:**

```plaintext
show ip pim bsr
```

Lists the identity, configuration, and time data of the currently elected BSR for the domain, plus the BSR-candidate configuration, the C-RP configuration, and the supported multicast groups on the current router.

**Listing BSR data for the domain and the immediate router**

```
Switch(config)# show ip pim bsr
Status and Counters - PIM-SM Bootstrap Router Information

Elected BSR Address : 10.10.10.2
E-BSR Priority : 0
E-BSR Hash Mask Length : 30
E-BSR Up Time : 53 mins
Next Bootstrap Message : 08 secs

C-BSR Admin Status : This system is a Candidate-BSR
C-BSR Address : 10.10.10.1
C-BSR Priority : 0
C-BSR Hash Mask Length : 30
C-BSR Message Interval : 60
C-BSR Source IP VLAN : 100

C-RP Admin Status : This system is a C-RP
C-RP Address : 10.10.10.1
C-RP Hold Time : 150
C-RP Advertise Period : 60
C-RP Priority : 192
C-RP Source IP VLAN : 100

Group Address| Group Mask
-------------|-------------
192.168.1.0 | 255.255.255.0
192.168.2.0 | 255.255.255.0
```

**Listing non-default BSR configuration settings**

The `show running` command includes the current non-default BSR configuration settings on the router.
Non-default BSR configuration listing

```
Switch(config)# show running

Running configuration:
:
:
ip routing
snmp-server community "public" Unrestricted
vlan 1
  
  
vlan 120
  
  
ip multicast-routing
router rip
  exit
router pim
  [bsr-candidate
  bsr-candidate source-ip-vlan 120 ]
  bsr-candidate priority 1
  rp-candidate
  rp-candidate source-ip-vlan 120
  rp-candidate group-prefix 224.0.0.0 240.0.0.0
  rp-candidate hold-time 150
  exit
vlan 120
  ip rip 120.10.10.2
  ip pim-sparse
    ip-addr any
  exit
  exit
  
```

Example of Non-Default BSR Candidate Configuration in the Router’s Running Configuration

**Note:** priority appears only if it is configured to a non-default value.

Displaying the current RP set

The BSR sends periodic RP updates to all C-RPs in the domain. These updates include the set of multicast group data configured on and reported by all C-RPs in the domain. This data does not include any static RP entries configured on any router in the domain. (To view the static RP-set information for any static RPs configured on a particular router, you must access the CLI of that specific router.)

**Syntax:**

```
show ip pim rp-set [learned | static]
```

Without options, this command displays the multicast group support for both the learned C-RP assignments and any statically configured RP assignments.

- **learned**
  - Displays only the learned C-RP assignments the router has learned from the latest BSR message.

- **static**
  - Displays only the statically configured RP assignment(s) configured on the router.

**Listing both the learned and static RP-set data**
Displaying only the learned RP-set data for the PIM-SM domain

```
switch(config)# show ip pim rp-set learned
Status and Counters - PIM-SM Learned RP-Set Information
Group Address  Group Mask  RP Address  Hold Time  Expire Time
231.100.128.0  255.255.240.0  100.10.10.1  150     150
232.240.255.252 255.255.255.255 100.10.10.1  150     150
237.255.248.1  255.255.255.255  100.10.10.1  150     150
239.10.10.240 255.255.255.252  120.10.10.2  150     150
239.10.10.240 255.255.255.252  120.10.10.2  150     150
```

Displaying only the static RP-set data (applies to current router only)

```
switch(config)# show ip pim rp-set static
Status and Counters - PIM-SM Static RP-Set Information
Group Address  Group Mask  RP Address  Override
231.100.128.255 255.255.255.255 100.10.10.1  Yes
```

Displaying C-RP data

Displaying the router’s C-RP status and configuration

Syntax:
show ip pim rp-candidate [config]
rp-candidate
Lists the current C-RP status and, if the status is enabled for C-RP operation, includes the current C-RP configuration on the router.
rp-candidate config
Lists the current C-RP status and the current C-RP configuration on the router, regardless of whether C-RP operation is currently enabled.

Listing for a router that is not configured as a C-RP

```
switch(pim)# show ip pim rp-candidate
This system is not a Candidate-RP
```
Full C-RP configuration listing

```
Switch[pim]# show ip pim rp-candidate config

Status and Counters - PIM-SM C-RP Information

C-RP Admin Status : This system is not a C-RP
C-RP Address : 120.10.10.2
C-RP Hold Time : 150
C-RP Advertise Period : 60
C-RP Priority : 192
C-RP Source IP VLAN : 120

Group Address Group Mask
--------------------------
239.10.10.240 255.255.255.252
```

Listing non-default C-RP configuration settings

The `show running` command includes the current non-default C-RP configuration settings on the router.

Non-default C-RP configuration listing

```
Switch(config)# show running

Running configuration:
.
.
ip routing
mp-sharp server community "public" Unrestricted
vlan 1
.
.
vlan 120
.
.
ip multicast-routing
router rip
  exit
router pim
  bsr-candidate
  bsr-candidate source-ip-vlan 120
  bsr-candidate_priority 1
  rp-candidate
  rp-candidate source-ip-vlan 120
  rp-candidate group-prefix 224.0.0.0 240.0.0.0
  rp-candidate hold-time 150
  exit
vlan 120
  ip rip 120.10.10.2
  ip pim-sparse
    ip-addr any
    .
    .
```

PIM-SM overview

In a network where IP multicast traffic is transmitted for multimedia applications, such traffic is blocked at routed interface (VLAN) boundaries unless a multicast routing protocol is running. Protocol Independent Multicast (PIM) is a family of routing protocols that form multicast trees to forward traffic from multicast sources to subnets that have used a protocol such as IGMP to request the traffic. PIM relies on the unicast routing tables created by any of several unicast routing protocols to identify the path back to a multicast source (reverse path forwarding, or...
RPF.) With this information, PIM sets up the distribution tree for the multicast traffic. The PIM-DM and PIM-SM protocols on the switches covered in this guide enable and control multicast traffic routing.

IGMP provides the multicast traffic link between a host and a multicast router running PIM-SM. Both PIM-SM and IGMP must be enabled on VLANs whose member ports have directly connected hosts with a valid need to join multicast groups.

PIM-DM is used in networks where, at any given time, multicast group members exist in relatively large numbers and are present in most subnets. However, using PIM-DM in networks where multicast sources and group members are sparsely distributed over a wide area can result in unnecessary multicast traffic on routers outside the distribution paths needed for traffic between a given multicast source and the hosts belonging to the multicast group. In such networks, PIM-SM can be used to reduce the effect of multicast traffic flows in network areas where they are not needed. And because PIM-SM does not automatically flood traffic, it is a logical choice in lower bandwidth situations.

**PIM-SM features**

PIM-SM on the switches covered in this guide include:

**Routing protocol support**

PIM uses whichever IP unicast routing protocol is running on the router. These can include:

- RIP
- OSPF
- Static routes
- Directly connected interfaces

**VLAN interface support:**

Up to 127 outbound VLANs (and 1 inbound VLAN) are supported in the multicast routing table (MRT) at any given time. This means the sum of all outbound VLANs across all current flows on a router may not exceed 127. (A single flow may span one inbound VLAN and up to 127 outbound VLANs, depending on the VLAN memberships of the hosts actively belonging to the flow.) For 2930F switch, 16 VLANs are supported.

**Flow capacity:**

Up to 2046 flows are supported in hardware across a maximum of 128 VLANs. (A flow is composed of an IP source address and an IP multicast group address, regardless of the number of active hosts belonging to the multicast group at any given time.)

**Multicast group to RP mapping:**

PIM-SM uses the BSR protocol to automatically resolve multicast group addresses to C-RP routers. In the current software release, a router administers BSR operation on a PIM-SM domain basis. (BSR zones and PIM border router operation are not currently supported by the switches covered in this guide.) Note that BSR operation does not extend to statically configured RPs.

**IGMP compatibility:**

PIM-SM is compatible with IGMP version 2, and is fully interoperable with IGMP for determining multicast flows.

**VRRP:**

PIM-SM is fully interoperable with VRRP to quickly transition multicast routes in the event of a failover.

**MIB support on the switches covered in this guide:**

PIM-SM supports the Protocol Independent Multicast MIB for IPv4 (RFC 2934.)

With some exceptions, PIM-SM supports the parts of the multicast routing MIB (RFC 2932) applicable to PIM-SM operation.

**PIM draft specifications:**

Compatible with PIM-SM specification (RFC 4061.)

**BSR implementation:**
Complies with RFC 5059 (scope zones are not supported.)

**Maximum number of routes:**
For the 2930F switch, the maximum number of routes is limited to 200.

### PIM-SM operation and router types

Unlike PIM-DM, PIM-SM assumes that most hosts do not want to receive multicast traffic, and uses a non-flooding multicast model to direct traffic for a particular multicast group from the source to the VLAN(s) where there are multicast receivers that have joined the group. As a result, this model sends traffic only to the routers that specifically request it.

#### PIM-SM operation

In a given PIM-SM domain, routers identified as DRs, RPs, and a BSR participate in delivering multicast traffic to the IP multicast receivers that request it. This approach avoids the flooding method of distributing multicast traffic (employed by PIM-DM) and is best suited for lower bandwidth situations.

The software supports the following operation to enable multicast traffic delivery within a PIM-SM domain:

- From a pool of eligible DR candidates in each VLAN, one DR is elected for each VLAN interface having at least one PIM-SM router. In a multinetted domain, this DR supports multicast traffic from a source on any subnet in the VLAN.
- From a pool of eligible BSR candidates in the domain, one BSR is elected for the entire domain.
- From a pool of eligible C-RPs, one is elected to support each multicast group or range of groups allowed in the domain, excluding any group supported only by static RPs. The multicast groups allowed in the domain are determined by the aggregation of the groups allowed by the individually configured RPs and any static RPs. (Note that RP-Cs and static RP’s can be configured with overlapping support for a given set of multicast groups.)

### Rendezvous-point tree (RPT)

When a DR in a VLAN receives traffic for a particular multicast group from a source on that VLAN, the DR encapsulates the traffic and forwards it to the RP elected to support that multicast group. The RP decapsulates the traffic and forwards it toward the multicast receiver(s) requesting that group. This forms an RPT extending from the DR through any intermediate PIM-SM routers leading to the PIM-SM edge router(s) for the multicast receiver(s) requesting the traffic. (If the RP has no current join requests for the group, the traffic is dropped at the RP.)

*Example PIM-SM domain with RPT active to support a host joining a multicast group*
Shortest-path tree (SPT)

SPTs are especially useful in high data-rate applications where reducing unnecessary traffic concentrations and throughput delays are significant. In the default PIM-SM configuration, SPT operation is automatically enabled. (The software includes an option to disable SPT operation.

Shortest-path tree operation

In the default PIM-SM configuration, after an edge router receives the first packet of traffic for a multicast group requested by a multicast receiver on that router, it uses Reverse Path Forwarding (RPF) to learn the shortest path to the group source. The edge router then stops using the RPT and begins using the shortest path tree (SPT) connecting the multicast source and the multicast receiver. In this case, when the edge router begins receiving group traffic from the multicast source through the SPT, it sends a prune message to the RP tree to terminate sending the requested group traffic on that route. (This results in entries for both the RP path and the STP in the routing table.) When completed, the switchover from the RPT to a shorter SPT can reduce unnecessary traffic concentrations in the network and reduce multicast traffic throughput delays.

Note that the switchover from RPT to SPT is not instantaneous. For a short period, packets for a given multicast group may be received from both the RPT and the SPT. Also, in some topologies, the RPT and the SPT to the same edge router may be identical.

Example PIM-SM domain with SPT active to support a host that has joined a multicast group

Restricting multicast traffic to RPTs
An alternate method to allowing the domain to use SPTs is to configure all of the routers in the domain to use only RPTs. However, doing so can increase the traffic load in the network and cause delays in packet delivery.

**Maintaining an active route for multicast group members**

The edge router itself and any intervening routers on the active tree between the members (receivers) of a multicast group and the DR for that group, send periodic joins. This keeps the active route available for as long as there is a multicast receiver requesting the group. When a route times out or is pruned, the DR ceases to send the requested group traffic on that route.

**Border routers and multiple PIM-SM domains**

Creating multiple domains enables a balancing of PIM-SM traffic within a network. Defining PIM-SM domain boundaries requires the use of PIM border routers (PMBRs), and multiple PMBRs can be used between any two domains.

---

**NOTE**

The software described in this guide does not support PMBR operation for PIM-SM networks.

---

**PIM-SM DT**

PIM-SM and DT can be enabled within the same VLAN. Considered the following conditions when enabling PIM-SM DT.

- To enable PIM-SM DT, configure PIM-SM and DT in the same VLAN.
- PIM-SM DT is not supported on a switch with v1 modules. The command `no allow-v1-modules` can be used to disable any v1 modules.
- Since there can be multiple combinations of DT and PIM-SM configured in multiple VLANs, PIM-SM DT feature should be enabled on the first combination of PIM-SM & DT in same VLAN and disabled when the last such pair is un-configured.

For information about the `show distributed-trunking consistency-parameters global feature pim-sm` and the `no allow-v1-module` commands, see the *Management and Configuration Guide* for your switch.

**PIM-SM router types**

Within a PIM-SM domain, PIM-SM routers can be configured to fill one or more of the roles described in this section.

**DR:**

A router performing this function forwards multicast traffic from a unicast source to the appropriate distribution (rendezvous) point.

**BSR:**

A router elected to this function keeps all routers in a PIM-SM domain informed of the currently assigned RP for each multicast group currently known in the domain.

**RP:**

A router elected as a RP for a multicast group receives requested multicast traffic from a DR and forwards it toward the multicast receiver(s) requesting the traffic. See RP.

**Static RP (static RP):**

This option forwards traffic in the same way as an RP, but requires manual configuration on all routers in the domain to be effective.

All of the above functions can be enabled on each of several routers in a PIM-SM domain.
DR
In a VLAN populated by one or more routers running PIM-SM, one such router is elected the DR for that VLAN. When the DR receives a Join request from a multicast receiver on that VLAN, it forwards the join toward the router operating as the RP for the requested multicast group.

Where multiple PIM-SM routers exist in a VLAN, the following criteria is used to elect a DR:

1. The router configured with the highest DR priority in the VLAN is elected.
2. If multiple routers in the VLAN are configured with the highest DR priority, the router having the highest IP address is elected.

In a given domain, each VLAN capable of receiving multicast traffic from a unicast source should have at least one DR. (Enabling PIM-SM on a VLAN automatically enables the router as a DR for that VLAN.) Because there is an election process for DR on each VLAN, all routers on a VLAN need to be enabled for DR. Where it is important to ensure that a particular router is elected as the DR for a given VLAN, you can increase the DR priority on that VLAN configuration for that router.

If it is necessary to prevent a router from operating as a DR on a given VLAN, disable DR operation by configuring the DR priority as zero (0.)

BSR
Before a DR can forward encapsulated packets for a specific multicast group to an RP, it must know which router in the domain is the elected RP for that multicast group. The BSR function enables this operation by doing the following:

Procedure

1. Learns the group-to-RP mappings on the C-RPs in the domain by reading the periodic advertisements each one sends to the BSR.
2. Distributes the aggregate C-RP information as an RP-set to the PIM-SM routers in the domain. This is followed by an election to assign a specific multicast group or range of groups to the C-RPs in the domain. (The software supports assignment of up to four multicast addresses and/or ranges of multicast addresses to a C-RP.)

The BSR periodically sends bootstrap messages to the other PIM-SM routers in the domain to maintain and update the RP-set data throughout the domain, and to maintain its status as the elected BSR.

---

**NOTE**

| Where static RPs are configured in the domain to support the same multicast group(s) as one or more (dynamic) C-RPs, then the RP-set data has the precedence for assigning RPs for these groups unless the static RPs have been configured with the override option and if the multicast group mask for the static RP equals or exceeds the same mask for the applicable C-RP(s). |

---

**BSR configuration and election**

There should be multiple BSR candidates configured in a PIM-SM domain so that if the elected BSR becomes unavailable, another router will take its place. In the BSR election process, the BSR candidate configured with the highest priority number is selected. Where the highest priority setting is shared by multiple candidates, the
candidate having the highest IP address is selected. In the event that the selected BSR subsequently fails, another election takes place among the remaining BSR candidates. To facilitate a predictable BSR election, configure a higher priority on the router you want elected as the BSR for the domain.

A router serving as the BSR for a domain should be central to the network topology. This helps to ensure optimal performance and also reduce the possibility of a network problem isolating the BSR.

BSR role in fault recovery

If the hold-time maintained in the BSR for a given C-RP’s latest advertisement expires before being refreshed by a new advertisement from the C-RP, the non-reporting C-RP is removed from the domain. In this case, the removed C-RP’s multicast groups are re-assigned to other C-RPs. (If no other C-RPs or static RPs in the domain are configured to support a multicast group from the non-reporting C-RP, that group becomes unavailable in the domain.)

RP

Instead of flooding multicast traffic as is done with PIM-DM, PIM-SM uses a set of multiple routers to operate as RPs. Each RP controls multicast traffic forwarding for one or more multicast groups as follows:

- Receives traffic from multicast sources (S) via a DR.
- Receives multicast joins from routers requesting multicast traffic.
- Forwards the requested multicast traffic to the requesting routers.

Note that the routers requesting multicast traffic are either edge routers or intermediate routers. Edge routers are directly connected to specific multicast receivers using ICMP to request traffic. Intermediate routers are on the path between edge routers and the RP. This is known as a RP Tree (RPT) where only the multicast address appears in the routing table. For example:

(*, G), where:

* = a variable (wildcard) representing the IP address of any multicast source

G = a particular multicast group address.

The software supports up to 100 RPs in a given PIM-SM domain.

Defining supported multicast groups

An RP in the default candidate configuration supports the entire range of possible multicast groups. This range is expressed as a multicast address and mask, where the mask defines whether the address is for a single address or a range of contiguous addresses:

<table>
<thead>
<tr>
<th>Multicast address</th>
<th>Mask</th>
<th>Address range</th>
</tr>
</thead>
<tbody>
<tr>
<td>224.0.0.0</td>
<td>240.0.0.0</td>
<td>224.0.0.0 - 239.255.255</td>
</tr>
</tbody>
</table>

An alternate way to express the above (default) address and mask is:

224.0.0.0/4

In non-default candidate configurations, an RP allows up to four ranges of contiguous multicast groups, and/or individual multicast groups, or both. For example:
If a given multicast group is excluded from all RPs in a given domain, then that group will not be available to the multicast receivers connected in the domain.

For more on this topic, see Configuring C-RPs on PIM-SM routers.

**C-RP election**

Within a PIM-SM domain, different RPs support different multicast addresses or ranges of multicast addresses. (That is, a given PIM-SM multicast group or range of groups is supported by only one active RP, although other C-RPs can also be configured with overlapping or identical support.)

A C-RP’s group-prefix configuration identifies the multicast groups the RP is enabled to support.

If multiple C-RPs have group-prefixes configured so that any of these RPs can support a given multicast group, then the following criteria are used to select the RP to support the group:

1. The C-RP configured with the longest group-prefix mask applicable to the multicast group is selected to support the group. Step 2 of this procedure applies if multiple RP candidates meet this criterion.

2. The C-RP configured with the highest priority is selected. Step 3 of this procedure applies if multiple RP candidates meet this criterion.

3. A hash function (using the configured `bsr-candidate hash-mask-length value`) generates a series of mask length values that are individually assigned to the set of eligible C-RPs. If the hash function matches a single RP candidate to a longer mask length than the other candidates, that candidate is selected to support the group. Apply step 4 of this procedure if the hash function matches the longest mask length to multiple RP candidates.

4. The C-RP having the highest IP address is selected to support the group.
In a PIM-SM domain where there are overlapping ranges of multicast groups configured on the C-RPs, discrete ranges of these groups are assigned to the domain’s C-RPs in blocks of sequential group numbers. The number of multicast groups in the blocks assigned within a given domain is determined by the

\[
\text{bsr-candidate hash-mask-length}
\]

value (range=1 to 32) configured on the elected BSR for the domain. A higher value means fewer sequential group numbers in each block of sequential group numbers, which results in a wider dispersal of multicast groups across the C-RPs in the domain.

As indicated above, multiple C-RPs can be configured to support the same multicast group(s). This is the generally recommended practice and results in redundancy that helps to prevent loss of support for desired multicast groups in the event that a router in the domain becomes unavailable.

Configuring a C-RP to support a given multicast group does not ensure election of the C-RP to support that group unless the group is excluded from all other RPs in the domain.

Also, within a PIM-SM domain, a router can be configured as a C-RP available for a given multicast group or range of groups and as the static RP for a given multicast group or range of groups. The recommended practice is to use C-RPs for all multicast groups unless there is a need to ensure that a specific group or range of groups is always supported by the same routing switch. See Static RP (static RP).

**Redundant Group Coverage Provides Fault-Tolerance**

If a C-RP elected to support a particular multicast group or range of groups becomes unavailable, the router is excluded from the RP-set. If the multicast group configuration of one or more other C-RPs overlaps the configuration in the failed RP, then another C-RP is elected to support the multicast group(s) formerly relying on the failed RP.

**Static RP (static RP)**

**General application**

Like C-RPs, static RPs control multicast forwarding of specific multicast groups or ranges of contiguous groups. However, static RPs are not dynamically learned, and increase the configuration and monitoring effort needed to maintain them. As a result, static RPs are not generally recommended for use except where one of the following conditions applies:

- It is desirable to designate a specific router interface as a backup RP for specific group(s.)
- Specific multicast groups are expected, and a static RP would help to avoid overloading a given RP with a high volume of multicast traffic.
- A C-RP for the same group(s) is less reliable than another RP that would not normally be elected to support the group(s.)
- Tighter traffic control or a higher priority is desired for specific multicast groups
While the use of C-RPs and a BSR enable a dynamic selection of RPs for the multicast group traffic in a network, using static RPs involves manually configuring all routers in the domain to be aware of each static RP. This can increase the possibility of multicast traffic failure from to misconfigurations within the PIM-SM domain. Also, because a BSR does not administer static RPs, troubleshooting PIM-SM traffic problems can become more complex. For these reasons, use of static RPs should be limited to applications where no viable alternatives exist, or where the network is stable and requires configuring and maintaining only a few routers.

If a static RP operating as the primary RP for a multicast group fails, and the PIM-SM configuration in the domain does not include a (secondary) dynamic RP (C-RP) backup to the static RP, then new multicast groups assigned to the static RP will not be available to multicast receivers in the domain. Also, if a static RP fails, support for existing groups routed through SPTs that exclude the failed router will continue, but any existing flows routed through the RPT will fail.

**Supporting a static RP as primary**

A static RP can be configured to operate as either a secondary or primary RP. With the primary option, a dynamic (C-RP) backup is recommended. The precedence of a static RP over a dynamic RP is determined by the following static RP configuration options:

- override enabled on the static RP.
- A group mask on the static RP that equals or exceeds the group mask on the C-RP for the same multicast group (s.)

For override configuration information, see [Statically configuring an RP to accept multicast traffic](#).

**Operating rules for static RPs**

- Static RPs can be configured on the same routers as C-RPs.
- Where a C-RP and a static RP are configured to support the same multicast group(s), the C-RP takes precedence over the static RP unless the static RP is configured to override the C-RP. (See Supporting a static RP as primary.)
- Any static RP in a domain must be configured identically on all routers in the domain. Otherwise, some DRs will not know of the static RP and will not forward the appropriate multicast traffic, and some routers will not know where to send Joins for the groups supported by static RP.
- Up to four static RP entries can be configured on a router. Each entry can be for either a single multicast group or a range of contiguous groups.
- Only one interface can be configured as the static RP for a given multicast group or range of groups. For example, a properly configured PIM-SM domain does not support configuring 10.10.10.1 and 10.20.10.1 to both support a multicast group identified as 239.255.255.10.
- Static RPs are not included in the RP-set messages generated by the BSR, and do not generate advertisements.
- If a static RP becomes unavailable, it is necessary to remove and/or replace the configuration for this RP in all routers in the domain.

**Configuration**

See [Statically configuring an RP to accept multicast traffic](#).
Operating rules and recommendations

Guideline for configuring C-RPs and BSRs

Routers in a PIM-SM domain should usually be configured as both C-RPs and candidate BSRs; this can reduce some overhead traffic.

The SPT policy should be the same for all RPs in a domain.

Allowing some RPs to remain configured to implement SPTs while configuring other RPs in the same domain to force RPT use can result in unstable traffic flows. (Use the `no ip pim-sparse spt-threshold` command to change between SPT and RPT operation on each router.)

Application of RPs to multicast groups.

In a PIM-SM domain, a given multicast group or range of groups can be supported by only one RP. (Typically, multiple C-RPs in a domain are configured with overlapping coverage of multicast groups, but only one such candidate will be elected to support a given group.)

Ensuring that the C-RPs in a PIM-SM domain cover all desired multicast groups.

All of the multicast groups you want to allow in a given PIM-SM domain must be included in the aggregate of the multicast groups configured in the domain's C-RPs. In most cases, all C-RPs in a domain should be configured to support all RP groups (the default configuration for a router enabled as a C-RP.) This provides redundancy in case an RP becomes unavailable. (If the C-RP supporting a particular multicast group becomes unavailable, another C-RP is elected to support the group as long as there is redundancy in the C-RP configuration for multiple routers.) Note that is cases where routers are statically configured to support a specific group or range of groups, the C-RP prioritization mechanism allows for redundant support.

PIM-SM and PIM-DM.

These two features cannot both be enabled on the same router at the same time.

Supporting PIM-SM across a PIM Domain.

To properly move multicast traffic across a PIM-SM domain, all routers in the domain must be configured to support PIM-SM. That is, a router without PIM-SM capability blocks routed multicast traffic in a PIM-SM domain.

Configuration steps for PIM-SM

This process assumes that the necessary VLANs and IP addressing have already been configured on the routing switch.

The switches described in this guide do not support PMBR operation in the current software release.

Planning considerations

- Where multiple routers are available to operate as the DR for a given source, set the DR priority on each router according to how you want the router used.
- Determine whether there are any bandwidth considerations that would call for disabling SPT operation. (If any routers in the domain have SPT operation disabled, it should be disabled on all RPs in the domain. See Operating rules for static RPs.)
- Determine the routers to configure as C-BSRs. In many applications, the best choice may be to configure all routers in the domain as candidates for this function.
- Determine the multicast group support you want on each C-RP and any static RPs in the domain. The easiest option is to enable C-RP to support all possible multicast groups on all routers in the domain. However, if there
are traffic control considerations you want to apply, you can limit specific multicast groups to specific routers and/or set priorities so that default traffic routes support optimum bandwidth usage.

**Per-router global configuration context**

Use these steps to enable routing and PIM operation in the global configuration context of each PIM-SM router (HP(config)#):

1. Enable routing. (Use `ip routing`.)
2. Enable multicast routing. (Use `ip multicast-routing`.)
3. Enable PIM. (Use `router pim enable`.)
4. Configure the routing method(s) needed to reach the interfaces (VLANs) on which you want multicast traffic available for multicast receivers in your network:
   - Enable RIP or OSPF. (Use `router rip enable` or `router ospf enable`)
   - If desired, configure static routes to the destination subnets. (Use `ip route dest-ip-address/mask-bits next-hop-ip-addr`.)

**Per-VLAN PIM-SM configuration**

These steps configure PIM-SM in the VLAN interface context for each VLAN configured on the router (switch (vlan-vid)#).

1. Enable IGMP. (Use `ip igmp`) Repeat this action on every router (and switch) having membership in the VLAN.

   ![Note](image)

   You can use either IGMPv2 or v3. Ensure that PIM and IGMP are enabled for the VLAN in the PIM designated router, even if no clients are connected.

2. For both the global and VLAN levels on the routers where there are connected multicast receivers that may issue joins or send multicast traffic, use the same routing method as Step 4 of this procedure.

3. Enable PIM-SM on the VLAN interfaces where you want to allow routed multicast traffic. (Default: disabled)

   a. If these VLANs do not already have static IP addresses, then statically configure one or more IP addresses on each VLAN you want to support PIM-SM operation. (PIM-SM cannot be enabled on a VLAN that does not have a statically configured IP address. That is, PIM-SM cannot use an IP address acquired by DHCP/Bootp.)

   b. Use `ip pim-sparse` to enter the VLAN’s `pim-sparse` context and do one of the following:
      - Enable PIM-SM on the VLAN and allow the default option (any) to dynamically determine the source IP address for the PIM-SM packets sent from this VLAN interface.
      - Enable PIM-SM on the VLAN and allow the default option (any) to dynamically determine the source IP address for the PIM-SM packets sent from this VLAN interface.
      - Enable PIM-SM on the VLAN and specify an IP address for the PIM-SM packets sent from this VLAN interface. (The specified IP address must already be statically configured on the VLAN.)
This step requires enabling Router PIM on the global configuration context. See Configuring global context commands.

**c. Option:** Change the current DR priority, in the PIM Sparse context, to a value for the current router in the current VLAN by using Command `dr-priority [0-4294967295]` (DR Priority default = 1)

When you initially enable PIM-SM, Hewlett Packard Enterprise recommends that you leave the PIM-SM traffic control settings at their default settings. You can then assess performance and make configuration changes when needed.

**4. Option:** Change one or more of the traffic control settings for the pim-sparse of a given VLAN on which PIM-SM is enabled. (Note that some VLAN context control settings apply to both PIM-SM and PIM-DM).

<table>
<thead>
<tr>
<th>Features accessed in VLAN - vid - pim - sparse context</th>
<th>Operation</th>
</tr>
</thead>
<tbody>
<tr>
<td>ip-addr</td>
<td>Sets or resets the source IP address for PIM-SM packets sent out on the interface. Also enables PIM-SM on the interface. (Default: any)</td>
</tr>
<tr>
<td>hello-interval1</td>
<td>Resets the interval between transmitted PIM Hello packets on the interface. (Default: 30 seconds)</td>
</tr>
<tr>
<td>hello-delay<strong>hello-interval</strong></td>
<td>Resets the maximum delay for transmitting a triggered PIM Hello packet on the interface. (Default: 5 seconds)</td>
</tr>
<tr>
<td>lan-prune-delay<strong>hello-interval</strong></td>
<td>Enables or disables the LAN prune delay feature on the interface. (Default: on)</td>
</tr>
<tr>
<td>override-interval<strong>hello-interval</strong></td>
<td>Resets the override interval of the LAN prune delay configured on the interface. (Default: 2500 milliseconds)</td>
</tr>
<tr>
<td>propagation-delay<strong>hello-interval</strong></td>
<td>Resets the delay interval for triggering LAN prune delay packets on the interface. (Default: 500 milliseconds)</td>
</tr>
<tr>
<td>dr-priority</td>
<td>Resets the priority of the interface in the Designated Router election process. (Default: 1)If you want one router on a given VLAN to have a higher priority for DR than other routers on the same VLAN, use the <code>dr-priority</code> command to reconfigure the DR priority setting as needed. Otherwise, the highest DR priority among multiple routers on the same VLAN interface is assigned to the router having the highest source IP address for PIM-SM packets on that interface.</td>
</tr>
</tbody>
</table>

**Router Pim configuration**

1. Applies to both PIM-SM and PIM-DM operations.
These steps configure the PIM-SM in the Router PIM context (switch (pim) #).

1. Specify the VLAN interface to advertise as the BSR Candidate and enable the router to advertise itself as a candidate BSR in a PIM-SM domain. (Use bsr-candidate source-ip-vlan vid.)

2. **Option:** To make NSR candidate selection occur quickly and predictably, set a different priority on each BSR candidate in the domain. (Use bsr-candidate priority.)

3. Do one of the following to configure RP operation:
   a. **Recommended**
      : Enable C-RP operation and configure the router to advertise itself as a C-RP to the BSR for the current domain. This step includes the option to allow the C-RP to be a candidate for either all possible multicast groups or for up to four multicast groups and/or ranges of groups. (Use rp-candidate source-ip-vlan vid [group-addr/group-mask].)
   b. **Option:** Use rp-address ip-addr [group-addr/group-mask] to statically configure the router as the RP for a specific multicast group or range of multicast groups. (This must be configured on all RIM-SM routers in the domain.)
   c. **Option:** Use rp-address ip-addr [group-addr/group-mask] to statically configure the router as the RP for a specific multicast group or range of multicast groups. (This must be configured on all RIM-SM routers in the domain.)

4. **Option:** In the PIM router context, change one or more of the traffic control settings.

**Table 11: Options Accessed in Router PIM Context**

<table>
<thead>
<tr>
<th>Options Accessed in Router PIM Context</th>
<th>Operation</th>
</tr>
</thead>
<tbody>
<tr>
<td>rp-candidate group-prefix group-addr/group-mask</td>
<td>Enter an address and mask to define an additional multicast group or a range of groups.</td>
</tr>
<tr>
<td>rp-candidate hold-time 30-255</td>
<td>Tells the BSR how long it should expect the sending C-RP router to be operative. (Default: 150; 0 if router is not a candidate.)</td>
</tr>
<tr>
<td>rp-candidate priority 0-255</td>
<td>Changes the priority for the C-RP router. When multiple C-RPs are configured for the same multicast group(s), the priority determines which router becomes the RP for such groups. A smaller value means a higher priority. (Default: 192)</td>
</tr>
<tr>
<td>[ no ] spt-threshold</td>
<td>Disable or enable the router’s ability to switch multicast traffic flows to the shortest path tree. (Default: enabled)</td>
</tr>
<tr>
<td>join-prune-interval 5-65535</td>
<td>Option: Globally change the interval for the frequency at which join and prune messages are forwarded on the router’s VLAN interfaces. (Default: 60 seconds)</td>
</tr>
<tr>
<td>trap</td>
<td>Option: Enable or disable PIM traps. (Default: disabled)</td>
</tr>
<tr>
<td>neighbor-loss</td>
<td>hardware-mrt-full</td>
</tr>
</tbody>
</table>
Operating notes
Overview of IP routing

The switches offer the following IP routing features:

**Static routes**

Up to 256 static routes

**RIP (Router Information Protocol)**

Supports RIP Version 1, Version 1 compatible with Version 2 (default), and Version 2

**OSPF (open shortest path first)**

The standard routing protocol for handling larger routed networks.

**IRDP (ICMP Router Discovery Protocol)**

Advertises the IP addresses of the routing interfaces on this switch to directly attached host systems

**DHCP Relay**

Allows you to extend the service range of your DHCP server beyond its single local network segment

---

Throughout this chapter, the switches are referred to as "routing switches." When IP routing is enabled on your switch, it behaves just like any other IP router.

---

**NOTE**

Basic IP routing configuration consists of adding IP addresses, enabling IP routing, and enabling a route exchange protocol, such as RIP.

---

To configure the IP addresses, see the Management and Configuration Guide for your switch.

---

Viewing the IP route table

The IP route table is displayed by entering the CLI command `show ip route` from any context level in the console CLI. Here is an example of an entry in the IP route table:

<table>
<thead>
<tr>
<th>Destination</th>
<th>Gateway</th>
<th>VLAN Type</th>
<th>Sub-Type</th>
<th>Metric</th>
<th>Di</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.10.10.1/32</td>
<td>10.10.12.1</td>
<td>connected</td>
<td></td>
<td>1</td>
<td></td>
</tr>
</tbody>
</table>

---

Increasing ARP age timeout (CLI)

The address resolution protocol (ARP) age is the amount of time the switch keeps a MAC address learned through ARP in the ARP cache. The switch resets the timer to zero each time the ARP entry is refreshed and removes the entry if the timer reaches the ARP age.

**Syntax:**

```plaintext
no ip arp-age [(1...1440) | infinite]
```

Allows the ARP age to be set from 1 to 1440 minutes (24 hours).
If the option `infinite` is configured, the internal ARP age timeout is set to 99,999,999 seconds (approximately 3.2 years). An `arp-age` value of 0 (zero) is stored in the configuration file to indicate that `infinite` has been configured. This value also displays with the `show` commands and in the menu display (Switch Configuration IP Config).

Default: 20 minutes

**Setting the ARP age timeout to 1000 minutes**

`switch(config)# ip arp-age 1000`

**Show IP command displaying ARP age**

To view the value of ARP age timer, enter the `show ip` command. The Arp Age time value is shown in bold below.

```
switch(config)# show ip
Internet (IP) Service
   IP Routing : Disabled
   Default Gateway : 15.255.120.1
   Default TTL : 64
   **Arp Age** : 1000
   Domain Suffix : DNS server :
VLAN | IP Config IP Address Subnet Mask Proxy ARP
--------------------- + --------------------- + --------------------- + --------------------- +
DEFAULT_VLAN | Manual 15.255.111.13 255.255.248.0 No
```

**IP ARP-age value in the running config file**

You can also view the value of the ARP age timer in the configuration file. The `ip arp-age 1000` value is shown in bold below.

```
switch(config)# show running-config
Running configuration:
; J9091A Configuration Editor; Created on release #K.15.XX
hostname "8200LP"
module 2 type J8702A
module 3 type J8702A
module 4 type J8702A
ip default-gateway 15.255.120.1
   **ip arp-age 1000**
snmp-server community "public" Unrestricted
snmp-server host 16.180.1.240 "public"
vlan 1
   name "DEFAULT_VLAN"
   untagged B1-B24,C1-C24,D1-D24
   ip address 15.255.120.85 255.255.248.0
exit
gvrp
spanning-tree
```

**Setting and viewing the arp-age value**

You can set or display using the menu interface (Switch Configuration IP Config).

**Displaying the ARP age value**
Reconfiguring the router ID (optional)

If you want to change the router ID setting, do the following:

1. Go to the global config context; the CLI prompt appears similar to the following:
   
   switch(config)#

2. If OSPF is not enabled, go to step on page 111; if OSPF is enabled, use `no router ospf` to disable OSPF operation.

3. Use `ip router-id ip-addr` to specify a new router ID. (This IP address must be unique in the routing switch configuration.)

4. If you disabled OSPF operation in step on page 111, use `router ospf` to re-enable OSPF operation.

For more information on the router ID, see IP global parameters for routing switches on page 124 and Changing the router ID on page 127.

Changing the router ID

**Syntax:**

```
ip router-id ip-addr
```

The `ip-addr` can be any valid, unique IP address.

```
switch(config)# ip router-id 209.157.22.26
```

**Note:** You can specify an IP address used for an interface on the routing switch, but do not specify an IP address in use by another device.

Enabling proxy ARP

Proxy ARP is disabled by default on routing switches. Enter the following commands from the VLAN context level in the CLI to enable proxy ARP:

```
switch(config)# vlan 1
switch(vlan-1)# ip proxy-arp
```

To again disable IP proxy ARP, enter:
Syntax:
no ip proxy-arp

Enabling local proxy ARP

When the local proxy ARP option is enabled, a switch responds with its MAC address to all ARP request on the VLAN. All IP packets are routed through and forwarded by the switch. The switch prevents broadcast ARP requests from reaching other ports on the VLAN.

Internet control message protocol (ICMP) redirects are disabled on interfaces on which local proxy ARP is enabled.

To enable local proxy ARP, you must first enter VLAN context, for example:

```
switch(config) vlan 1
```

Then enter the command to enable local proxy ARP:

```
switch(vlan-1)ip local-proxy-arp
```

Syntax:
no ip local-proxy-arp

Enables the local proxy ARP option. You must be in VLAN context to execute this command.

When enabled on a VLAN, the switch responds to all ARP requests received on the VLAN ports with its own hardware address.

The `no` option disables the local proxy ARP option.

Default: Disabled

Execute the `show ip` command to see which VLANs have local proxy ARP enabled.

Local proxy ARP is enabled on the default VLAN

```
switch(vlan-1)# show ip

Internet (IP) Service
IP Routing : Disabled
Default TTL : 64
Arp Age : 20
Domain Suffix :
DNS server :

<table>
<thead>
<tr>
<th>VLAN</th>
<th>IP Config</th>
<th>IP Address</th>
<th>Subnet Mask</th>
<th>Proxy ARP</th>
</tr>
</thead>
<tbody>
<tr>
<td>DEFAULT_VLAN</td>
<td>DHCP/Bootp</td>
<td>15.255.157.54</td>
<td>255.255.248.0</td>
<td>Yes Yes</td>
</tr>
<tr>
<td>VLAN2100</td>
<td>Disabled</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
```

Configuring source MAC based ARP attack detection (ARP throttle)

ARP throttle operation
Source-MAC based ARP attack detection (ARP throttle) protects the switch CPU from ARP attacks by enabling restriction of the overall number of ARP packets the CPU receives from a given client. An ARP attack occurs when the switch receives more ARP packets from the same source MAC address than allowed by the configured threshold setting. ARP throttle uses a “remediation mode” to determine whether to simply monitor the frequency of ARP packets or actually restrict the ARP packet traffic from a given client. In cases where a device in your network is sending a large quantity of ARP packets for legitimate purposes, you can configure ARP throttling to exclude that device from being monitored.

When enabled in the default configuration, ARP throttle:

- monitors incoming ARP packets and “allowlists” clients sending excessive ARP packets to the switch
- maintains a count of clients sending ARP packets to the switch

When configured to filter ARP packet traffic, ARP throttle monitors ARP packet traffic as described above, and also drops ARP packets received from allowlisted clients.

Non-default ARP throttle settings persist when ARP throttle is disabled.

`ip arp-throttle enable`

This command enables or disables ARP throttle operation for monitoring or filtering of ARP packets received by the switch from other devices. (Default: disabled.) Enabling ARP-throttling uses the currently configured settings to immediately invoke ARP attack monitoring and (if configured), to filter ARP packet traffic from devices transmitting excessive ARP packets.

**Syntax**

```
ip arp-throttle enable
no ip arp-throttle enable
```

**Description**

Enables ARP throttle operation. The `no` form of the command disables ARP throttle operation.

`ip arp-throttle remediation-mode`

Determines the disposition of ARP packets the switch receives.

**Syntax**

```
ip arp-throttle remediation-mode <monitor | filter>
```

When ARP throttle is enabled in `monitor` mode (the default), the switch does the following:

- Monitors ARP packet traffic received by the switch CPU.
- Assigns “allowlist” status to devices generating an excessive numbers of ARP packets within a five-second period.
- Maintains a running total of the devices from which ARP packets are being received.

When ARP throttle is enabled in `filter` mode, the switch drops all ARP packet traffic received from allowlisted devices while continuing to perform the above three `monitor` actions.

**Example**

Configure the switch to drop ARP packet traffic received from allowlisted devices.

```
switch(config)# ip arp-throttle remediation-mode filter
```

`ip arp-throttle aging-time`

Configures the time in seconds that a allowlisted device remains on the allowlist. (Default: 300 seconds.) If the switch is configured to filter ARP packets as described above, then the ARP packets received from allowlisted devices are dropped.

**Syntax**
ip arp-throttle aging-time <1-86400>

Example
Configure the switch to reinstate allowlisted clients after 600 seconds on the allowlist.
switch(config)# ip arp-throttle aging-time 600

ip arp-throttle threshold
Specifies the number of ARP packets per five-second period that the switch can receive from another device. (Default: 30.) Exceeding this rate places the source device on the allowlist. If the switch is configured to filter ARP packets as described for remediation mode (page yy), then the ARP packets received from allowlisted devices are dropped.

Syntax
ip arp-throttle threshold <1 - 1024>

Example
Configure the switch to allowlist a client from which it receives more than eight ARP packets in a five second period.
switch(config)#ip arp-throttle threshold 8

ip arp-throttle exclude-mac
Excludes traffic from a device having the specified MAC address from ARP packet monitoring and filtering, and adds the MAC address to the Excluded MAC List in the output for the show ip arp-throttle command (page xx). You can exclude up to ten MAC addresses.

Syntax
no ip arp-throttle exclude-mac <MAC-addr [MAC-addr...MAC-addr]>

Options
no
Where exclude-mac has been used to exclude traffic from a device having the specified MAC address for ARP packet monitoring and filtering, the no option restores ARP packet traffic from that device to IP ARP throttling, and removes the device MAC address from the Excluded MAC List.

Example
Exclude the clients having the following two MAC addresses from IP ARP-throttling, then use show ip arp-throttle to view the result in the Excluded MAC List:

- 001018-0158c8
- 01555d-c95d0a

switch(config)# ip arp-throttle exclude-mac 001018-0158c8 01555d-c95d0a
switch(config)# show ip arp-throttle
Source MAC Based ARP Attack Detection Information

- Enabled : Yes
- Remediation Mode : Filter
- Threshold (pkt) : 30
- Allowlist Age (sec) : 300

- Excluded MAC List
- 001018-0158c8
- 01555d-c95d0a

- Clients in Allowlist : 3
Clients Being Tracked : 190

Restore the client having the MAC address 001018-0158c8 to IP ARP-throttling and then use `show ip arp-throttle` to view the result in the Excluded MAC List:

```
switch(config)# no ip arp-throttle exclude-mac 001018-0158c8
switch(config)# show ip arp-throttle
```

```
Source MAC Based ARP Attack Detection Information

   Enabled : Yes
   Remediation Mode : Filter
   Threshold (pkt) : 30
   Allowlist Age (sec) : 300

Excluded MAC List
   ------------------
   01555d-c95d0a

Clients in Allowlist : 4
Clients Being Tracked : 189
```

**show ip arp-throttle**

This command shows the current ARP throttle configuration, excluded MAC list, and client statistics.

**Syntax**

```
show ip arp-throttle
```

**Example**

This output indicates ARP throttle is enabled, filtering ARP packets according to the default packet threshold and aging-time settings. ARP packets from a device identified as 000f20-aeaec0 are excluded from ARP throttling, and statistics indicate 4 allowlisted clients and the ARP packet traffic of 180 clients being tracked.

```
switch# show ip arp-throttle
```

```
Source MAC Based ARP Attack Detection Information

   Enabled : Yes
   Remediation Mode : Filter
   Threshold (pkt) : 30
   Allowlist Age (sec) : 300

Excluded MAC List
   ------------------
   000f20-aeaec0

Clients in Allowlist : 4
Clients Being Tracked : 180`
The "Clients in Allowlist" and "Clients being Tracked" counters shown above operate only when ARP throttle is enabled. Rebooting the switch restarts the counters from zero. Executing any of the following commands causes the switch to reset these counters to zero:

- `ip arp-throttle enable`
  (Starts the counters from zero.)
- `no ip arp-throttle enable`
  (Resets the counters to zero.)
- `ip arp-throttle remediation-mode <monitor | filter>`
  (Restarts the counters from zero if the `ip arp-throttle remediation-mode` setting is changed.)

If a failover occurs on a 5400R switch, the switch maintains the allowlist status of any currently allowlisted clients. However, the current list of tracked clients is cleared and restarted.

**Identifying allowlisted and restored clients**

The switch event log records an entry when `ip arp-throttle` allowlists a client, removes a client from the allowlist, or drops an ARP packet received from a allowlisted client. Use the show logging command to display entries for these actions.

**Example**

```
switch# show logging -r

  Keys:  W=Warning  I=Information
        M=Major     D=Debug   E=Error

    ---- Reverse event Log listing: Events Since Boot ----

W 02/16/16 22:57:16 02539 arpt: ST1-CMDR: Client 20fdf1-e0935b exceeds the limit of ARP packets and is allowlisted.

W 02/16/16 22:57:16 02541 arpt: ST1-CMDR: An ARP packet from allowlist client 20fdf1-e0935b is dropped. (4 times in 60 seconds)

W 02/16/16 22:57:03 02539 arpt: ST1-CMDR: Client d0bf9c-13c149 exceeds the limit of ARP packets and is allowlisted.

I 02/16/16 21:52:05 02540 arpt: ST1-CMDR: Client 20fdf1-e0935b is moved out of allowlist due to inactivity.
```

**Enabling forwarding of IP directed broadcasts (CLI)**

For more information, see Configuring forwarding parameters.

```
switch(config)# ip directed-broadcast
```

**Syntax:**

```
no ip directed-broadcast
```
Te switch software makes the forwarding decision based on the routing switch's knowledge of the destination network prefix. Routers cannot determine that a message is unicast or directed broadcast apart from the destination network prefix. The decision to forward or not forward the message is by definition only possible in the last-hop router.

### Introduction to feature

Wake-on-LAN (WOL) is an Ethernet computer networking standard that allows a computer to be turned on or awakened by a network message. The message is sent by a program executed on the same local area network. Messages can also be initiated from another network by using subnet directed broadcasts or a WOL gateway service. WOL is implemented using specially designed packet called magic packet. WOL is enabled on the switch by using a `ip directed-broadcast` command with an IPv4 configuration, which can be used to specify an access-list name, thus avoiding unnecessary administrative overhead.

IP directed-broadcasts would only be forwarded if permitted by the associated access-list. An `implicit deny` at the end of an access list drops all IP directed-broadcasts that are not authorized according to the access list entries.

**NOTE**

IP routing must be enabled on the switch for this feature to work.

### CLI commands

The optional association of access-list with IP directed-broadcast allows user to filter directed broadcast traffic alone based on access-list entry rule. The feature’s CLI includes an optional parameter to specify access-list name along with the already existing “ip directed-broadcast” command. The access-list rule specified is applied globally on the switch and is not specific to any vlan's alone. There is an Implicit Deny at the end of an access list that will drop all IP Directed Broadcasts that do not match any of the access list entries.

### Configuration commands

Enable IP directed broadcast forwarding for Wake-on-LAN support. An optional ACL can also be applied to control what packets are forwarded.

**Syntax**

```
switch(config)# ip directed-broadcast [access-group <ACL-ID>]
```

**Apply the specified access control list.**

```
access-list-name-str
```

**ASCII string specifying an ACL**

**Example configuration**

```
switch(config)# ip directed-broadcast [access-group] <wol-acl>
```

**Example running configuration**

```
; J9573A Configuration Editor; Created on release #K14.18.0000x
; Ver #06:7c ff.ff.ff.3f.ef:57
hostname "switch-name"
module 1 type j9573x
ip access-list extended "wol-acl"
....10 permit ip 192.168.1.10 0.0.0.0 182.168.1.1 0.0.0.255
....exit

ip directed-broadcast access-group "wol-acl"
ip routing
```
snmp-server community "public" unrestricted
  ....ip address dhcp-bootp
  ....exit
  vlan 1
  ....name "DEFAULT_VLAN"
  ....no untagged 1,23-24
  ....untagged 2-22,25-26
  ....ip address dhcp-bootp
  ....exit
  vlan 10
  ....name "VLAN10"
  ....untagged 1
  ....ip address 192.168.1.1 255.255.255.0
  ....exit
  vlan 20
  ....name "VLAN20"
  ....untagged 23-24
  ....ip address 182.168.1.1 255.255.255.0
  ....exit

Figure 6  Configuration diagram
If specified ACL ID is non-existing, it is not possible to associate with IP Directed Broadcast. An error will be shown to the user.

It is not allowed to delete an ACL which is associated with IP Directed Broadcast and on attempt, an error message will be shown to user.

The same ACL wol-ACL can be applied to any other interface like VLAN, port and tunnel.

### <wol-ACL> entries

```
ip access-list extended <wol-ACL>
10 permit ip 192.168.1.1 255.255.255.0 182.168.1.1 55.255.255.0
20 deny ip 172.168.1.1 255.255.255.0 162.168.1.1 255.255.255.0
Exit
```

### Show commands

IP directed broadcast hit counts for the associated access-list with can be displayed using the `show` command.

### Show statistics

#### Show IPV4 ACL Statistics.

**Syntax**

```
switch # show statistics aclv4 <acl-id>
```

**Options**

- `port <port>`
- `vlan <vlan-id>`
- `ip-directed-broadcast`

**Reset IPV4 Statistics.**

Please note that the existing help text of all other parameters listed other than newly added `ip-directed-broadcast` will remain the same.

**Syntax**

```
show statistics aclv4 <acl-name-str>
```

#### ip-directed-broadcast

Clear Statistics for the IP Directed Broadcast ACL.

**Syntax**

```
clear statistics aclv4 <acl-id>
```

**Options**

- `port <port>`
Reset IPV4 Statistics.

Please note that the existing help text of all other parameters listed other than newly added ip-directed-broadcast will remain the same.

Syntax

clear statistics aclv4 <acl-name-str>

ip-directed-broadcast Clear Statistics for the IP Directed Broadcast ACL.

show access-list command

The existing “show access-list” command will have the following modification to support ip-directed-broadcast.

Syntax

show access-list

Options

ACL-ID> [config]
<config >
<ip-directed-broadcast>
ports <PORT-LIST>
radius
resources

Show Access Control List Information.

Please note that the existing help of all other parameters listed other than newly added ip-directed-broadcast will remain the same.

Show ACL’s applied to IP Directed Broadcast traffic

show access-list <ip-directed-broadcast>

switch # show access-list ip-directed-broadcast

Access Lists for IP Directed Broadcast
IPv4 : wol-acl  Type: Extended

If user uses already existing show access-list <ACL_NAME-STR> command, the status of ACL on IP Directed Broadcast will be shown applied as in this example below.

switch # sh access-list wol-acl
Access Control Lists
......Name: wol-acl
......Type: Extended
......Applied: Yes
......SEQ  Entry
-----------------------------------------------------------------------
10 .Action: permit
......Src IP: 192.168.1.1  Mask: 255.255.255.0  Port(s):
MIB

MIB object hpifcDBroadcastFwdAcl stores the access-list name associated with IP directed broadcast.

- **hpifcDBroadcastFwdEnable**
  
  **Syntax** integer
  
  - enabled (1)
  - disabled (2)
  - MAX-ACCESS read-write
  
  STATUS current

  Used to enable/disable IP directed broadcast feature on the device. When set to disable, hpifcDBroadcastFwdAcl is also cleared.

- **hpifcDBroadcastFwdAcl**
  
  **Syntax** integer
  
  SnmpAdminString (SIZE (1..64))
  
  MAX-ACCESS read-write
  
  STATUS current

  Used to store the access-list name associated with the IP Directed Broadcast feature. This is a printable string up to 64 characters in size and case sensitive. An empty string indicates that no access-list is associated with the IP directed broadcast feature. This object can be configured only when the value of the object hpifcDBroadcastFwdEnable is set to enable.

Disabling the directed broadcasts

```
switch(config)# no ip directed-broadcast
```

Disabling replies to broadcast ping requests

By default, devices are enabled to respond to broadcast ICMP echo packets, which are ping requests. (See Disabling ICMP messages.) You can disable response to ping requests on a global basis using the following CLI command:
switch(config)# no ip icmp echo broadcast-request

Syntax:
no ip icmp echo broadcast-request

If you need to re-enable response to ping requests, enter the following command:

switch(config)# ip icmp echo broadcast-request

Disabling all ICMP unreachable messages
For more information, see Disabling ICMP destination unreachable messages.

switch(config)# no ip icmp unreachable

Syntax:
no ip icmp unreachable

Disabling ICMP redirects
You can disable ICMP redirects on the routing switch only on a global basis, for all the routing-switch interfaces.
Enter the following command at the global CONFIG level of the CLI:

switch(config)# no ip icmp redirects

Syntax:
no ip icmp redirects

IP interfaces
On the routing switches, IP addresses are associated with individual VLANs. By default, there is a single VLAN (Default_VLAN) on the routing switch. In that configuration, a single IP address serves as the management access address for the entire device. If routing is enabled on the routing switch, the IP address on the single VLAN also acts as the routing interface.

Each IP address on a routing switch must be in a different subnet. You can have only one VLAN interface in a given subnet. For example, you can configure IP addresses 192.168.1.1/24 and 192.168.2.1/24 on the same routing switch, but you cannot configure 192.168.1.1/24 and 192.168.1.2/24 on the same routing switch.

You can configure multiple IP addresses on the same VLAN.
The number of IP addresses you can configure on an individual VLAN interface is 32.
You can use any of the IP addresses you configure on the routing switch for Telnet, Web management, or SNMP access, as well as for routing.

All devices support configuration and display of IP address in classical subnet format (example: 192.168.1.1 255.255.255.0) and Classless Interdomain Routing (CIDR) format (example: 192.168.1.1/24). You can use either format when configuring IP address information. IP addresses are displayed in classical subnet format only.

IP tables and caches

ARP cache table
The ARP cache contains entries that map IP addresses to MAC addresses. Generally, the entries are for devices that are directly attached to the routing switch.

An exception is an ARP entry for an interface-based static route that goes to a destination that is one or more router hops away. For this type of entry, the MAC address is either the destination device’s MAC address or the MAC address of the router interface that answered an ARP request on behalf of the device, using proxy ARP.

**ARP cache**

The ARP cache contains dynamic (learned) entries. The software places a dynamic entry in the ARP cache when the routing switch learns a device’s MAC address from an ARP request or ARP reply from the device.

The software can learn an entry when the switch or routing switch receives an ARP request from another IP forwarding device or an ARP reply. Here is an example of a dynamic entry:

**ARP cache dynamic entry**

<table>
<thead>
<tr>
<th>IP Address</th>
<th>MAC Address</th>
<th>Type</th>
<th>Port</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 207.95.6.102</td>
<td>0800.5afc.ea21</td>
<td>Dynamic</td>
<td>6</td>
</tr>
</tbody>
</table>

Each entry contains the destination device’s IP address and MAC address.

To configure other ARP parameters, see [Configuring ARP parameters](#).

**IP route table**

The IP route table contains routing paths to IP destinations.

The default gateway, which you specify when you configure the basic IP information on the switch, is used only when routing is not enabled on the switch.

**Routing paths**

The IP route table can receive the routing paths from the following sources:

- Directly-connected destination, which means there are no router hops to the destination
- Static route, which is a user-configured route
- Route learned through RIP
- Route learned through OSPF

**Administrative distance**

The IP route table contains the best path to a destination. When the software receives paths from more than one of the sources listed above, the software compares the administrative distance of each path and selects the path with the lowest administrative distance. The administrative distance is a protocol-independent value from 1 to 255.

The IP route table is displayed by entering the `show ip route` command from any context level in the console CLI. Here is an example of an entry in the IP route table:

**IP route table entry**

<table>
<thead>
<tr>
<th>Destination</th>
<th>Gateway</th>
<th>VLAN Type</th>
<th>Sub-Type</th>
<th>Metric</th>
<th>Di</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.10.10.1/32</td>
<td>10.10.12.1</td>
<td>connected</td>
<td></td>
<td></td>
<td>1</td>
</tr>
</tbody>
</table>

Each IP route table entry contains the destination’s IP address and subnet mask and the IP address of the next-hop router interface to the destination. Each entry also indicates route type, and for OSPF routes, the subtype, and the route’s IP metric (cost). The type indicates how the IP route table received the route.
Enter the `show ip route summary` command to display the aggregate count of routes for each routing protocol.

### IP route summary display

```
switch(config)# show ip route summary

IPv4 Route Table Summary

+----------------+-----------+
| Protocol        | Active Routes |
|----------------+-----------|
| Connected       | 1         |
| Static          | 1         |
```

To configure a static IP route, see [Static Routing](#).

### IP forwarding cache

The IP forwarding cache provides a fast-path mechanism for forwarding IP packets. The cache contains entries for IP destinations. When a routing switch has completed processing and addressing for a packet and is ready to forward the packet, the device checks the IP forwarding cache for an entry to the packet’s destination.

- If the cache contains an entry with the destination IP address, the device uses the information in the entry to forward the packet out the ports listed in the entry. The destination IP address is the address of the packet’s final destination. The port numbers are the ports through which the destination can be reached.
- If the cache does not contain an entry, the software can create an entry in the forwarding cache.

Each entry in the IP forwarding cache has an age timer. The age interval depends on the number of entries in the table. The age timer ranges from 12 seconds (full table) to 36 seconds (empty table.) Entries are aged only if they are not being used by traffic. If you have an entry that is always being used in hardware, it will never age. If there is no traffic, it will age in 12 to 36 seconds. The age timer is not configurable.

---

**NOTE**

You cannot add static entries to the IP forwarding cache.

### IP route exchange protocols

The switch supports the following IP route exchange protocols:

- Routing Information Protocol (RIP)
- Open Shortest Path First (OSPF).
- ICMP Router Discovery Protocol (IRDP)
- Dynamic Host Configuration Protocol (DHCP) Relay

These protocols provide routes to the IP route table. You can use one or more of these protocols, in any combination. The protocols are disabled by default. For configuration information, see the following:

- Configuring RIP parameters on page 143
- Configuring IRDP on page 256
- Dynamic Host Configuration Protocol on page 259

### IP global parameters for routing switches

The following table lists the IP global parameters and the page where you can find more information about each parameter.
### Table 12: IP global parameters for routing switches

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
<th>Default</th>
<th>See page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Router ID</td>
<td>The value that routers use to identify themselves to other routers when exchanging route information. OSPF uses the router ID to identify routers. RIP does not use the router ID.</td>
<td>The lowest-numbered IP address configured on the lowest-numbered routing interface.</td>
<td>Changing the router ID</td>
</tr>
<tr>
<td>Address Resolution Protocol (ARP)</td>
<td>A standard IP mechanism that routers use to learn the MAC address of a device on the network. The router sends the IP address of a device in the ARP request and receives the device's MAC address in an ARP reply.</td>
<td>Enabled</td>
<td>Configuring ARP parameters</td>
</tr>
<tr>
<td>ARP age</td>
<td>The amount of time the device keeps a MAC address learned through ARP in the device's ARP cache. The device resets the timer to zero each time the ARP entry is refreshed and removes the entry if the timer reaches the ARP age. See Increasing ARP age timeout (CLI).</td>
<td>Five minutes.</td>
<td>N/A</td>
</tr>
<tr>
<td>Proxy ARP</td>
<td>An IP mechanism a router can use to answer an ARP request on behalf of a host, by replying with the router's own MAC address instead of the host's.</td>
<td>Disabled</td>
<td>About enabling proxy ARP</td>
</tr>
<tr>
<td>Time to Live (TTL)</td>
<td>The maximum number of routers (hops) through which a packet can pass before being discarded. Each router decreases a packet's TTL by 1 before forwarding the packet. If decreasing the TTL causes the TTL to be 0, the router drops the packet instead of forwarding it.</td>
<td>64 hops</td>
<td>See the Management and Configuration Guide for your switch.</td>
</tr>
<tr>
<td>Directed broadcast forwarding</td>
<td>A directed broadcast is a packet containing all ones (or in some cases, all zeros) in the host portion of the destination IP address. When a router forwards such a broadcast, it sends a copy of the packet out each of its enabled IP interfaces. You also can enable or disable this parameter on an individual interface basis. See IP interface parameters for routing switches.</td>
<td>Disabled</td>
<td>Enabling forwarding of directed broadcasts</td>
</tr>
<tr>
<td>ICMP Router Discovery Protocol (IRDP)</td>
<td>An IP protocol that a router can use to advertise the IP addresses of its router interfaces to directly attached hosts. You can enable or disable the protocol at the Global CLI Config level. You also can enable or disable IRDP and configure the following protocol parameters on an individual VLAN interface basis at the VLAN Interface CLI Config level.</td>
<td>Disabled</td>
<td>A-21 A-159</td>
</tr>
<tr>
<td>Parameter</td>
<td>Description</td>
<td>Default</td>
<td>See page</td>
</tr>
<tr>
<td>----------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>--------------------------</td>
<td>----------</td>
</tr>
<tr>
<td></td>
<td>Forwarding method (broadcast or multicast)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Hold time</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Maximum advertisement interval</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Minimum advertisement interval</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Router preference level</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Static route</td>
<td>An IP route you place in the IP route table.</td>
<td>No entries</td>
<td>A-25</td>
</tr>
<tr>
<td>Default network route</td>
<td>The router uses the default network route if the IP route table does not</td>
<td>None configured</td>
<td>A-30</td>
</tr>
<tr>
<td></td>
<td>contain a route to the destination. Enter an explicit default route (0.0.0.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.0.0.0 or 0.0.0.0/0) as a static route in the IP route table.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**IP interface parameters for routing switches**

The following table lists the interface-level IP parameters for routing switches.

**Table 13: IP interface parameters – routing switches**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
<th>Default</th>
<th>See page</th>
</tr>
</thead>
<tbody>
<tr>
<td>IP address</td>
<td>A Layer 3 network interface address; separate IP addresses on individual VLAN interfaces.</td>
<td>None configured</td>
<td>1</td>
</tr>
<tr>
<td>Metric</td>
<td>A numeric cost the router adds to RIP routes learned on the interface. This parameter applies only to RIP routes.</td>
<td>1 (one)</td>
<td>A-33</td>
</tr>
<tr>
<td>ICMP Router Discovery Protocol (IRDP)</td>
<td>Locally overrides the global IRDP settings.</td>
<td>Disabled</td>
<td>A-159</td>
</tr>
<tr>
<td>IP helper address</td>
<td>The IP address of a UDP application server (such as a BootP or DHCP server) or a directed broadcast address. IP helper addresses allow the routing switch to forward requests for certain UDP applications from a client on one subnet to a server on another subnet.</td>
<td>None configured</td>
<td>A-164</td>
</tr>
</tbody>
</table>

1 See the Management and Configuration Guide for your switch.
Configuring IP parameters for routing switches

The following sections describe how to configure IP parameters. Some parameters can be configured globally and overridden for individual VLAN interfaces. Other parameters can be configured on individual VLAN interfaces.

For IP configuration information when routing is not enabled, see the Management and Configuration Guide for your switch.

Configuring IP addresses

You can configure IP addresses on the routing switch's VLAN interfaces. For more information, see the Management and Configuration Guide for your switch.

Changing the router ID

Syntax:

```
ip router-id ip-addr
```

The `ip-addr` can be any valid, unique IP address.

```
switch(config)# ip router-id 209.157.22.26
```

You can specify an IP address used for an interface on the routing switch, but do not specify an IP address in use by another device.

Configuring ARP parameters

ARP is a standard IP protocol that enables an IP routing switch to obtain the MAC address of another device's interface when the routing switch knows the IP address of the interface. ARP is enabled by default and cannot be disabled.

How ARP works

A routing switch needs to know a destination's MAC address when forwarding traffic, because the routing switch encapsulates the IP packet in a Layer 2 packet (MAC layer packet) and sends the Layer 2 packet to a MAC interface on a device directly attached to the routing switch. The device can be the packet's final destination or the next-hop router toward the destination.

The routing switch encapsulates IP packets in Layer 2 packets regardless of whether the ultimate destination is locally attached or is multiple router hops away. Since the routing switch's IP route table and IP forwarding cache contain IP address information but not MAC address information, the routing switch cannot forward IP packets based solely on the information in the route table or forwarding cache. The routing switch needs to know the MAC address that corresponds with the IP address of either the packet's locally attached destination or the next-hop router that leads to the destination.

For example, to forward a packet whose destination is multiple router hops away, the routing switch must send the packet to the next-hop router toward its destination, or to a default route or default network route if the IP route table does not contain a route to the packet's destination. In each case, the routing switch must encapsulate the packet and address it to the MAC address of a locally attached device, the next-hop router toward the IP packet's destination.
To obtain the MAC address required for forwarding a datagram, the routing switch does the following:

- First, the routing switch looks in the ARP cache (not the static ARP table) for an entry that lists the MAC address for the IP address. The ARP cache maps IP addresses to MAC addresses. The cache also lists the port attached to the device and, if the entry is dynamic, the age of the entry. A dynamic ARP entry enters the cache when the routing switch receives an ARP reply or receives an ARP request (which contains the sender’s IP address and MAC address.) A static entry enters the ARP cache from the static ARP table (which is a separate table) when the interface for the entry comes up. To ensure the accuracy of the ARP cache, each dynamic entry has its own age timer. The timer is reset to zero each time the routing switch receives an ARP reply or ARP request containing the IP address and MAC address of the entry. If a dynamic entry reaches its maximum allowable age, the entry times out and the software removes the entry from the table. Static entries do not age-out and can be removed only by you.

- If the ARP cache does not contain an entry for the destination IP address, the routing switch broadcasts an ARP request out all of its IP interfaces. The ARP request contains the IP address of the destination. If the device with the IP address is directly attached to the routing switch, the device sends an ARP response containing its MAC address. The response is a unicast packet addressed directly to the routing switch. The routing switch places the information from the ARP response into the ARP cache. ARP requests contain the IP address and MAC address of the sender, so all devices that receive the request learn the MAC address and IP address of the sender and can update their own ARP caches accordingly.

The ARP request broadcast is a MAC broadcast, which means the broadcast goes only to devices that are directly attached to the routing switch. A MAC broadcast is not routed to other networks. However, some routers, including routing switches, can be configured to reply to ARP requests from one network on behalf of devices on another network. For more information, see About enabling proxy ARP.

If the routing switch receives an ARP request packet that it is unable to deliver to the final destination because of the ARP time-out, and no ARP response is received (the routing switch knows of no route to the destination address), the routing switch sends an ICMP Host Unreachable message to the source.

### About enabling proxy ARP

Proxy ARP allows a routing switch to answer ARP requests from devices on one network on behalf of devices in another network. Since ARP requests are MAC-layer broadcasts, they reach only the devices that are directly connected to the sender of the ARP request. Thus, ARP requests do not cross routers.

For example, if Proxy ARP is enabled on a routing switch connected to two subnets, 10.10.0/24 and 20.20.0/24, the routing switch can respond to an ARP request from 10.10.0.69 for the MAC address of the device with IP address 20.20.0.69. In standard ARP, a request from a device in the 10.10.0/24 subnet cannot reach a device in the 20.20.0 subnet if the subnets are on different network cables, and thus is not answered.

An ARP request from one subnet can reach another subnet when both subnets are on the same physical segment (Ethernet cable), since MAC-layer broadcasts reach all the devices on the segment.

### Proxy ARP and local proxy ARP behavior

When local proxy ARP is enabled, all valid ARP requests receive a response.

When proxy ARP is enabled, all valid ARP requests receive a response if the following conditions are met:

- There is a route to the target IP address in the ARP request (this can be a route or default route), and the VLAN (interface) the ARP request is received on does **NOT** match the interface for the next hop in the matched route to get to the target IP address.

**AND**
There is a route back to the source IP address in the ARP request and the interface the ARP request came in on does match the interface for the nex thop in the matched route to get to the source IP address.

### Configuring forwarding parameters

The following configurable parameters control the forwarding behavior of routing switches:

- **Time-To-Live (TTL) threshold**
  For more information, see the Management and Configuration Guide for your switch.
- **Forwarding of directed broadcasts**

These parameters are global and thus affect all IP interfaces configured on the routing switch.

### Enabling forwarding of directed broadcasts

A directed broadcast is an IP broadcast to all devices within a single directly-attached network or subnet. A net-directed broadcast goes to all devices on a given network. A subnet-directed broadcast goes to all devices within a given subnet.

---

**Note:** A less common type, the all-subnets broadcast, goes to all directly-attached subnets. Forwarding for this broadcast type also is supported, but most networks use IP multicasting instead of all-subnet broadcasting.

Forwarding for all types of IP directed broadcasts is disabled by default. You can enable forwarding for all types if needed. You cannot enable forwarding for specific broadcast types.

### Configuring ICMP

You can configure the following ICMP limits:

- **Burst-normal**
  The maximum number of ICMP replies to send per second.
- **Reply limit**
  You can enable or disable ICMP reply rate limiting.

### Disabling ICMP messages

The devices are enabled to reply to ICMP echo messages and send ICMP Destination Unreachable messages by default.

You can selectively disable the following types of Internet Control Message Protocol (ICMP) messages:

- **Echo messages (ping messages)**
  The routing switch replies to IP pings from other IP devices.
- **Destination unreachable messages**
  If the routing switch receives an IP packet that it cannot deliver to its destination, the routing switch discards the packet and sends a message back to the device that sent the packet to the routing switch. The message informs the device that the destination cannot be reached by the routing switch.
- **Address mask replies**
  You can enable or disable ICMP address mask replies.

### Disabling ICMP destination unreachable messages
By default, when a device receives an IP packet that the device cannot deliver, the device sends an ICMP unreachable message back to the host that sent the packet. The following types of ICMP unreachable messages are generated:

**Administration**
The packet was dropped by the device due to a filter or ACL configured on the device.

**Fragmentation-needed**
The packet has the "Don't Fragment" bit set in the IP Flag field, but the device cannot forward the packet without fragmenting it.

**Host**
The destination network or subnet of the packet is directly connected to the device, but the host specified in the destination IP address of the packet is not on the network.

**Network**
The device cannot reach the network specified in the destination IP address of the packet.

**Port**
The destination host does not have the destination TCP or UDP port specified in the packet. In this case, the host sends the ICMP Port Unreachable message to the device, which in turn sends the message to the host that sent the packet.

**Protocol**
The TCP or UDP protocol on the destination host is not running. This message is different from the Port Unreachable message, which indicates that the protocol is running on the host but the requested protocol port is unavailable.

**Source-route-failure**
The device received a source-routed packet but cannot locate the next-hop IP address indicated in the packet's Source-Route option.

---

**NOTE**
Disabling an ICMP Unreachable message type does not change the device's ability to forward packets. Disabling ICMP Unreachable messages prevents the device from generating or forwarding the Unreachable messages.
This chapter describes how to add static and null-routes to the IP route table.

**Configuring an IPv4 Route**

**Static route**
Configure a static route to a specific network or host address

**null-route**
Configure a "null" route to discard IP traffic to a specific network or host address:
- Discard traffic for the destination, with ICMP notification to sender
- Discard traffic for the destination, without ICMP notification to sender

**Syntax:**
```
no ip route dest-ip-addr / mask-length [next-hop-ip-addr | vlan vlan-id | reject | null-route] [metric metric] [distance1-255] [tag-value tagval] [name <name-str>]
```
Allows the addition and deletion of static routing table entries. A route entry is identified by a destination (IP address/mask length) and next-hop pair. The next-hop can be either a gateway IP address, a VLAN, or the keyword "reject" or "null-route".
A gateway IP address does not have to be directly reachable on one of the local subnets. If the gateway address is not directly reachable, the route is added to the routing table as soon as a route to the gateway address is learned.

- **dest-ip-addr / mask-bits**
The route destination and network mask length for the destination IP address. Alternatively, you can enter the mask itself
  For example, you can enter either 10.0.0.0/24 or 10.0.0.0 255.255.255.0 for a route destination of 10.0.0.0 255.255.255.0.

- **next-hop-ip-addr**
  This IP address is the gateway for reaching the destination. The next-hop IP address is not required to be directly reachable on a local subnet. (If the next-hop IP address is not directly reachable, the route will be added to the routing table as soon as a route to this address is learned.)

- **reject**
  Specifies a null-route where IP traffic for the specified destination is discarded and an ICMP error notification is returned to the sender.

- **null-route**
  Specifies a null-route where IP traffic for the specified destination is discarded and no ICMP error notification is returned to the sender.

- **distance**
  Specifies the administrative distance to associate with a static route. If not specified, this value is set to a default of 1. (Range: 1 to 255)
tag

Specifies a unique integer value for a given ECMP set (destination, metric, distance.)

name <name-str>

Assigns a name to a static route.

The `no` form of the command deletes the specified route for the specified destination next-hop pair.

**Examples**

*Configuring Names for Static Routes for IPv4*

```plaintext
Switch(config)# ip route 127.0.0.0/8 reject name Redroute
Switch(config)# ip route 11.0.0.0/24 10.10.10.5 name Blueroute
```

*Output Displaying Names of Static Routes*

```plaintext
switch# show ip route static name

   IP Route Entries
   ------------------- --------------- --------------- ---- --- -------
   Destination       Gateway       VLAN   Type   Name
   ------------------- --------------- --------------- ---- --- -------
      127.0.0.0/8       reject        Static   Redroute
      11.0.0.0/24       10.0.0.5      1      Static   Blueroute

```

*Output for a Specified Named Static Route*

```plaintext
switch# show ip route static name Redroute

   Destination      : 127.0.0.0/8
   Gateway          : reject
   VLAN             :
   Type             : static
   Sub-type         :
   Metric           :
   Distance         :
   Route Name       : Redroute

```

*Detailed Output of Named Static Routes*

```plaintext
switch# show ip route static name detail

   Destination      : 127.0.0.0/8
   Gateway          : reject
   VLAN             :
   Type             : static
   Sub-type         :
   Metric           :
   Distance         :
   Route Name       : Redroute

   Destination      : 11.0.0.0/24
   Gateway          : 10.0.0.5
   VLAN             : 1
   Type             : static
   Sub-type         :
   Metric           :
   Distance         :
   Route Name       : Blueroute

```

**Configuring an IPv6 Route**

**Syntax:**
no ipv6 route dest-ip-addr / prefix-length [next-hop-gateway-addr | vlan vid | tunnel tunnel-id|null-route|reject] [metric metric] [distance - 255] [tag-value tagval] [name <string>]
dest-ip-addr / prefix-length

The network prefix for the destination IPv6 address.

next-hop-gateway-addr|vlan <vid> | tunnel <tunnel-id>>

The gateway for reaching the destination. The next-hop address option (link-local or global unicast) is not required to be directly reachable on a local subnet. (If it is not directly reachable, the route will be added to the routing table when a path to this address is learned.) If the next-hop address is link-local, it must include both the address and the applicable VLAN VID or tunnel <tunnel-id>. For example: FE80::127%vlan10, where VLAN 10 is the interface where FE80::127 exists. For a tunnel, it would be FE80::127%tun3.

null-route

Specifies a null-route where IP traffic for the specified destination is discarded and no ICMP error notification is returned to the sender.

reject

Specifies a null-route where IP traffic for the specified destination is discarded and an ICMP error notification is returned to the sender.

metric

Specifies an integer value that is associated with the route. It is used to compare a static route to routes in the IP route table from other sources to the same destination.

distance

Specifies the administrative distance to associate with a static route. If not specified, this value is set to a default of 1. (Range: 1 to 255)

tag

Specifies a unique integer value for a given ECMP set (destination, metric, distance.)

name <name-str>

Assigns a name to a static route.
The no form of the command deletes the specified static or null-route from the routing table.

Examples

Configuring Names for Static Routes for IPv6

```
Switch(config)# ipv6 route 2001:400:2::/64 reject name Redroute
Switch(config)# ipv6 route 2001:300:2::/64 10.0.0.5 name Blueroute
```

Output for Unnamed Static Routes in IPv6
Switch(config)# show ipv6 route static

IPv6 Route Entries

T (Type):
S: Static  C: Connected  O: OSPFV3

ST (Sub-type):
O: OSPF Intra  E1: Externall  NI: NSSA Ext1
O1: OSPF Inter  E2: Externall2  NI2: NSSA Ext2

<table>
<thead>
<tr>
<th>Destination/Gateway</th>
<th>T</th>
<th>ST</th>
<th>Distance</th>
<th>Metric</th>
</tr>
</thead>
<tbody>
<tr>
<td>2001:400:2:1::/64</td>
<td>S</td>
<td>NA</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2001:300:2:1::/64</td>
<td>S</td>
<td>NA</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>10.0.0.5</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Output for Named Static Routes in IPv6

Switch# show ipv6 route static name

IPv6 Route Entries

<table>
<thead>
<tr>
<th>Destination/Gateway</th>
<th>T</th>
<th>Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>2001:400:2:1::/64</td>
<td>S</td>
<td>Redroute</td>
</tr>
<tr>
<td>2001:300:2:1::/64</td>
<td>S</td>
<td>Blueroute</td>
</tr>
<tr>
<td>10.0.0.5</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Output for a Specified Named Static Route in IPv6

Switch# show ipv6 route static name Redroute

<table>
<thead>
<tr>
<th>Destination</th>
<th>2001:400:2:1::/64</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gateway</td>
<td>reject</td>
</tr>
<tr>
<td>VLAN</td>
<td></td>
</tr>
<tr>
<td>Type</td>
<td>static</td>
</tr>
<tr>
<td>Sub-type</td>
<td></td>
</tr>
<tr>
<td>Metric</td>
<td>0</td>
</tr>
<tr>
<td>Distance</td>
<td>0</td>
</tr>
<tr>
<td>Route Name</td>
<td>Redroute</td>
</tr>
</tbody>
</table>

Detailed Output of Named Static Routes in IPv6

Switch# show ipv6 route static name detail

<table>
<thead>
<tr>
<th>Destination</th>
<th>2001:400:2:1::/64</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gateway</td>
<td>reject</td>
</tr>
<tr>
<td>VLAN</td>
<td></td>
</tr>
<tr>
<td>Type</td>
<td>static</td>
</tr>
<tr>
<td>Sub-type</td>
<td></td>
</tr>
<tr>
<td>Metric</td>
<td>0</td>
</tr>
<tr>
<td>Distance</td>
<td>0</td>
</tr>
<tr>
<td>Route Name</td>
<td>Redroute</td>
</tr>
<tr>
<td>Destination</td>
<td>2001:300:2:1::/64</td>
</tr>
<tr>
<td>Gateway</td>
<td>10.0.0.5</td>
</tr>
<tr>
<td>VLAN</td>
<td></td>
</tr>
<tr>
<td>Type</td>
<td>static</td>
</tr>
<tr>
<td>Sub-type</td>
<td></td>
</tr>
<tr>
<td>Metric</td>
<td>1</td>
</tr>
<tr>
<td>Distance</td>
<td>1</td>
</tr>
<tr>
<td>Route Name</td>
<td>Blueroute</td>
</tr>
</tbody>
</table>
Viewing static route information

The `show ip route static` command displays the current static route configuration on the routing switch. Configuring equal cost multi-path (ECMP) routing for static IP routes shows the configuration resulting from the static routes configured in the example above.

**Example:**

**Figure 7  Displaying the currently configured static routes**

```
Switch(config)# show ip route static

   IP Route Entries
Destination         Gateway      VLAN Type Sub-Type Metric Dist.
------------------- ----------- -------- ------- ------ ----
10.50.10.177/32     reject      static   1       1      1
10.10.40.0/24       VLAN10      10       static  1       1      1
10.50.10.25/27      VLAN10      10       static  1       1      1
10.50.10.0/24       blackhole  static   0       0      0
127.10.144.32/24    10.0.0.2   1        static 12      10     10
127.10.144.32/24    10.0.0.3   1        static 12      10     10

This reject (default null) route is included by default. Refer to "Configuring a static route" on page 1-1
An ECMP set with `ip load-sharing` set to 2 (the maximum paths allowed)
```

Configuring the default route

You can also assign the default route and enter it in the routing table. The default route is used for all traffic that has a destination network not reachable through any other IP routing table entry. For example, if 208.45.228.35 is the IP address of your ISP router, all non-local traffic could be directed to the ISP by entering this command:

```
switch(config)# ip route 0.0.0.0/0 208.45.228.35
```

Static route types

You can configure the following types of static IP routes:

**Standard**

The static route consists of a destination network address or host, a corresponding network mask, and the IP address of the next-hop IP address.

**Null-Route (discard)**

The null-route consists of the destination network address or host, a corresponding network mask, and either the `reject` or `null-route` keyword. Typically, the null-route is configured as a backup route for discarding traffic if the primary route is unavailable. By default, when IP routing is enabled, a route for the 127.0.0.0/8 network is created to the null interface. Traffic to this interface is rejected (dropped.)

This route is for all traffic to the "loopback" network, with the single exception of traffic to the host address of the switch’s loopback interface (127.0.0.1/32.) Figure A-3 on page 1-6 shows the default null-route entry in the switch's routing table.

On a single routing switch you can create one null-route to a given destination. Multiple null-routes to the same destination are not supported.

**Other sources of routes in the routing table**
The IP route table can also receive routes from the following sources:

- Directly connected networks: One route is created per IP interface. When you add an IP interface, the routing switch automatically creates a route for the network the interface is in.
- RIP: If RIP is enabled, the routing switch can learn about routes from the advertisements other RIP routers send to the routing switch. If the RIP route has a lower administrative distance than any other routes from different sources to the same destination, the routing switch places the route in the IP route table.
- OSPF: See RIP, but substitute "OSPF" for "RIP".
- Default route: This is a specific static route that the routing switch uses if other routes to the destination are not available.

**Static IP route parameters**

When you configure a static IP route, you must specify the following parameters:

- The IP address and network mask for the route's destination network or host.
- The route's path, which can be one of the following:
  - IP address of a next-hop router.
  - "Null" interface; the routing switch drops traffic forwarded to the null interface.

The routing switch also applies default values for the route's administrative distance (page A-10.) In the case of static routes, this is the value the routing switch uses to compare a static route to routes from other route sources to the same destination before placing a route in the IP route table. The default administrative distance for static IP routes is 1, but can be configured to any value from 1 to 255.

The fixed administrative distance values ensure that the routing switch always prefers static IP routes over routes from other sources to the same destination.

**Static route states follow VLAN states**

IP static routes remain in the IP route table only so long as the IP interface to the next-hop router is up. If the next-hop interface goes down, the software removes the static route from the IP route table. If the next-hop interface comes up again, the software adds the route back to the route table.

This feature allows the routing switch to adjust to changes in network topology. The routing switch does not continue trying to use routes on unreachable paths, but instead uses routes only when their paths are reachable.

For example, the following command configures a static route to 207.95.7.0 (with a network mask of 255.255.255.0), using 207.95.6.15 as the next-hop router's IP address.

```
switch(config)# ip route 207.95.7.0/24 207.95.6.15
```

A static IP route specifies the route's destination address and the next-hop router's IP address or routing switch interface through which the routing switch can reach the destination. (The route is added to the routing switch's IP route table.)

In the above example, routing switch "A" knows that 207.95.6.15 is reachable through port A2, and assumes that local interfaces within that subnet are on the same port. Routing switch "A" deduces that IP interface 207.95.7.188 is also on port A2. The software automatically removes a static route from the route table if the next-hop VLAN used by that route becomes unavailable. When the VLAN becomes available again, the software automatically re-adds the route to the route table.

**Configuring equal cost multi-path (ECMP) routing for static IP routes**

ECMP routing allows multiple entries for routes to the same destination. Each path has the same cost as the other paths, but a different next-hop router. The `ip load-sharing` command specifies the maximum number of equal paths that can be configured. Values range from 2 to 4.
Example of an ECMP set with the same destination but different next-hop routers

This example shows configuration of an ECMP set with two different gateways to the same destination address but through different next-hop routers. For more information, see IPv6 Configuration Guide for your switch.

```bash
switch(config)# ip route 127.10.144.21/24 10.10.10.2 metric 12 distance 10
switch(config)# ip route 127.10.144.21/24 10.10.10.3 metric 12 distance 10
```
Overview of Null-Route

Networking switches drop the received packet destined for null-routes without logging any packet information. Information like source IP, destination IP, VlanID and Port ID of the packets destined for the configured static null-routes is logged with this feature.

A static null-route is a route manually configured, on a switch, to drop packets destined for a particular network/IP address as it can be configured to a host as well as to a network. The user is able to set up the dropping of particular packets destined for a given IP address by enabling or disabling the route. Since all the packets received on the null-route are being logged, there is a performance impact and the exact performance numbers will be evaluated.

A new debug type is also introduced with this feature to log null debug messages.

Commands

**ip route**

Within the config context:

**Syntax**

```
ip route <IP-ADDR/MASK-LENGTH> null-route logging
ipv6 route <IPV6-ADDR/MASK-LENGTH> null-route logging
```

**Description**

Configures the debug logging for a static null-route for either IPv4 or IPv6.

**Options**

```
logging
```

Allows the packets received on the null-route to be logged. When logging is enabled on the switch for null-routes, the debug logs are sent to the configured destination. The destination can be a logging server, a buffer, or even the switch itself.

**Usage**

```
o ip route <IP-ADDR/MASK-LENGTH> null-route
no ipv6 route <IP-ADDR/MASK-LENGTH> null-route
```

Disables the logging facility for the configured null-routes.

The `no` form of the command does not have the logging option.

**Debug Logs**

The following debug message will be logged after a null-route logging is enabled.

```
Packet destined null-route with Source Ip =<IP ADDRESS>,
Destination Ip=<DESTINATION IP>, VLAN Interface = <VLAN INTERFACE>, Lport = <PORT ID>
```

No debug messages for null-route will be logged after a null-route logging is disabled.

```
ip route 20.20.20.2/32 null-route logging
```
Enable debug logging for a null-route with destination IP address.

```
Switch(config)# ip route 20.20.20.2/32 null-route logging
```

**ipv6 route 2001::2/128 null-route logging**

Enable debug logging for a null-route with destination IPv6 address.

```
switch(config)# ipv6 route 2001::2/128 null-route logging
```

**no ip route 20.20.20.2/32 null-route**

Disable debug logging for the null-route with destination IP address 20.20.20.2.

```
switch(config)# no ip route 20.20.20.2/32 null-route
```

**no ipv6 route 2001::2/128 null-route**

Disable debug logging for the null-route with destination IPv6 address

```
switch(config)#no ipv6 route 2001::2/128 null-route
```

**ip route 20.20.20.0/24 null-route logging**

Enable debug logging for IPv4 null-route 20.20.20.0

```
Switch(config)# ip route 20.20.20.0/24 null-route logging
```

**ipv6 route 2001::/64 null-route logging**

Enable debug logging for IPv6 null-route.

```
switch(config)#ipv6 route 2001::/64 null-route logging
```

**no debug ip fib null-route**

Within the config context:

**Syntax**
no debug ip fib null-route

**Description**
Enables debug logs for IPv4 null-routes.

**Parameters**
null-route

Enables debug logging of packets destined for null-routes.

**debug ip fib null-route**

Enable null-route logging.

```
switch(config)# debug ip fib null-route
switch(config) # debug ipv6 fib null-route
```

**no debug ip fib null-route**

Disable null-route logging.

```
switch(config)# no debug ip fib null-route
```

**Usage**
no debug ip fib null-route
no debug ipv6 fib null-route

**sys-debug ip fib null-route**

Within the config context.

**Syntax**

sys-debug ip fib null-route
no sys-debug ip fib null-route

**Description**

Enables user to make persistent null-route debug logging configuration across a reboot. Saves the configuration in the configuration tree and enables logging for any debug type. The command `sys-debug ip fib null-route` will enable the command `debug ip fib null-route` automatically. See example below.

**Parameters**

`null-route`  
Configures null-route debug logging for persistence.

**Usage**

no sys-debug ip fib null-route
no sys-debug ipv6 fib null-route

**Examples**

```bash
sys-debug ip fib null-route
no sys-debug ip fib null-route
```

Stack-switch-name(config)# show debug
  Debug Logging
    Source IP Selection: Outgoing Interface
    Origin identifier: Outgoing Interface IP
    Destination:
      Session
      Memory buffer
    Enabled debug types:
      None are enabled.
      debug for ip and ipv6 not enabled
    tty=none

Stack-switch-name(config)# sys-debug ip fib null-route
enabling using sys-debug command

Stack-switch-name(config)# sys-debug ipv6 fib null-route

Stack-switch-name(config)# show debug
  Debug Logging
    Source IP Selection: Outgoing Interface
    Origin identifier: Outgoing Interface IP
    Destination:
      Session
      Memory buffer
    Enabled debug types:
      ip fib null-route which is enabling debug and adding sys0debug config in running-config for persistence across reload.
      ipv6 fib null-route
    tty=none

Stack-switch-name(config)# show run

Running configuration:
  ; hpStack Configuration Editor; Created on release #KA.16.02.0000x
  ; Ver #0d:00.92.34.5f.3c.6b.fb.ff.ff.3f.ef:8a
  stacking
    member 1 type "J9586A" mac-address 5065f3-b4e200
    member 2 type "J9575A" mac-address 5cb901-26c0c0
    member 3 type "J9575A" mac-address 5cb901-26c880
  exit

hostname "Stack-switch-name"
Modifying existing commands

no sys-debug <FILTER-TYPE> | <FILTER-OPTIONS>

Syntax
sys-debug <FILTER-TYPE> | <FILTER-OPTIONS>
nosys-debug <FILTER-TYPE> | <FILTER-OPTIONS>

Description
Used to configure the type of messages displayed in the log.

Options
Filter-type
Either IP or IPv6.
Filter-options
Use the filter term null-route.

debug ipv6 fib null-route

Within the config context:

Syntax
debug ipv6 fib null-route

Description
Enable null-route logging

Usage
no debug ipv6 fib null-route

no sys-debug <DESTINATION> [logging | buffer]

Syntax
sys-debug <DESTINATION> [logging | buffer]

Description
Configures persistent debug logging. Used to configure the destination for the debug messages.

Options
logging
Send debug messages to the system event log and to any SYSLOG servers configured.

buffer
Send debug messages to a temporary buffer that is not saved across reboots.

no sys-debug ip fib null-route logging

Disable persistent null-route logging.

switch(config)# no sys-debug ip fib null-route
Show commands enhancement

show ip route

Syntax
show ip route

Description
Show if logging is enabled for the null-route.

show ip route

Switch# show ip route
IP Route Entries
            Destination   Gateway       VLAN Type Sub-Type Metric Dist.
------  -----------------  ---------  -------  ------  ------  ------
      20.20.20.2/32      null-route  static     1        1        1

show ipv6 route

show ipv6 route
Destination/
           Gateway
------  -----------------  ------  ------  -------  ------
      2001::2/128
           null-route

show running-config

Syntax
show running-config

Description
Displays all configured null-routes.

show running-config

Switch# show running-config
Running configuration:
ip route 20.20.20.20 255.255.255.255 null-route logging
ipv6 route 2001::2/128 null-route logging

Restrictions
- No support for web UI.
- No support for dynamic null-routes.
- No support for sampling or time-interval based logging. All packets matching the null-route would be logged.
- Performance and scale impact not considered.
Overview of RIP

Routing Information Protocol (RIP) is an IP route exchange protocol that uses a distance vector (a number representing distance) to measure the cost of a given route. The cost is a distance vector because the cost often is equivalent to the number of router hops between the routing switch and the destination network. A routing switch can receive multiple paths to a destination. The software evaluates the paths, selects the best path, and saves the path in the IP route table as the route to the destination. Typically, the best path is the path with the fewest hops. A hop is another router through which packets must travel to reach the destination. If the routing switch receives an RIP update from another router that contains a path with fewer hops than the path stored in the routing switch’s route table, the routing switch replaces the older route with the newer one. The routing switch then includes the new path in the updates it sends to other RIP routers, including routing switches. RIP routers, including routing switches, also can modify a route’s cost, generally by adding to it, to bias the selection of a route for a given destination. In this case, the actual number of router hops may be the same, but the route has an administratively higher cost and is thus less likely to be used than other, lower-cost routes. A RIP route can have a maximum cost of 15. Any destination with a higher cost is considered unreachable. Although limiting to larger networks, the low maximum hop count prevents endless loops in the network.

The switches support the following RIP types:

- Version 1
- V1 compatible with V2
- Version 2 (the default)

If the routing switch receives an ARP request packet that it is unable to deliver to the final destination because of the ARP timeout and no ARP response is received (the routing switch knows of no route to the destination address), the routing switch sends an ICMP Host Unreachable message to the source.

Configuring RIP parameters

Use the following procedures to configure RIP parameters on a system-wide and individual VLAN interface basis.

Enabling RIP

RIP is disabled by default. To enable it, use one of the following methods. When you enable RIP, the default RIP version is RIPv2-only. You can change the RIP version on an individual interface basis to RIPv1 or RIPv1-or-v2, if needed.

Syntax:

no router rip

To enable RIP on a routing switch, enter the following commands:

```
switch(config)# ip routing
switch(config)# router rip
```
Enabling RIP on the routing switch and entering the RIP router context

Syntax:
no router rip [[enable] | [disable]] [auto-summary]

Executed at the global configuration level to enable RIP on the routing switch and to enter the RIP router context. This enables you to proceed with assigning RIP areas and to modify RIP global parameter settings as needed. Global IP routing must be enabled before the RIP protocol can be enabled.

enable
Enables RIP routing.

disable
Disables RIP routing.

Default: Disabled

The no form of the command deletes all protocol-specific information from the global context and interface context. All protocol parameters are set to default values.

The no router rip command also disables RIP routing.

If you disable RIP, the switch retains all the configuration information for the disabled protocol in flash memory. If you subsequently restart RIP, the existing configuration will be applied.

The auto-summary form of the command enables advertisement of the summarized routes. When used with the no form of the command, auto-summary disables the advertisement of the summarized routes.

Example

Enter RIP router context

switch(config)# router rip
switch(rip)#

Enable RIP routing

switch(config)# router rip enable
switch(rip)#

Disable RIP routing

switch(config)# router rip disable
switch(rip)#

Delete all protocol-specific information from the global context and interface context and set all protocol parameters to default values

IP routing must be enabled prior to enabling RIP. The first command in the preceding sequence enables IP routing.
Deleting all protocol-specific information from the global and interface context sets all protocol parameters to default values.

IP routing must be enabled before enabling RIP. The first command in the preceding sequence enables IP routing.

To enable RIP on all IP addresses in a VLAN, use `ip rip` in the VLAN context. When the command is entered without specifying any IP address, it is enabled in all configured IP addresses of the VLAN.

To enable RIP on a specific IP address in a VLAN, use `ip rip [ip-addr | all]` in the VLAN context and enter a specific IP address. If you want RIP enabled on all IP addresses, you can specify `all` in the command instead of a specific IP address.

### Configuring a RIP authentication key

Configures a RIP authentication key. There is a maximum of 16 characters.

**Syntax:**

```
no ip rip [ip-addr] authentication-key key-string
```

When the switch is in enhanced secure mode, commands that take a secret key as a parameter have the echo of the secret typing replaced with asterisks. The input for `key-string` is prompted for interactively. For more information, see the Access Security Guide for your switch.

### Changing the RIP type on a VLAN interface

When you enable RIP on a VLAN interface, RIPv2-only is enabled by default. You can change the RIP type to one of the following on an individual VLAN interface basis:

- Version 1 only
- Version 2 only (the default)
- Version 1 - or - version 2

**Syntax:**

```
no ip rip [v1-only | v1-or-v2 | v2-only]
```

To change the RIP type supported on a VLAN interface, enter commands such as the following:

```
switch(config)# vlan 1
switch(vlan-1)# ip rip v1-only
```
Changing the cost of routes learned on a VLAN interface

By default, the switch interface increases the cost of an RIP route that is learned on the interface. The switch increases the cost by adding one to the route’s metric before storing the route.

You can change the amount that an individual VLAN interface adds to the metric of RIP routes learned on the interface.

RIP considers a route with a metric of 16 to be unreachable. Use this metric only if you do not want the route to be used. In fact, you can prevent the switch from using a specific interface for routes learned though that interface by setting its metric to 16.

To increase the cost a VLAN interface adds to RIP routes learned on that interface, enter commands such as the following:

```
switch(config)# vlan 1
switch(vlan-1)# ip rip metric 5
```

These commands configure vlan-1 to add 5 to the cost of each route learned on the interface.

**Syntax:**

```
ip rip metric 1-16
```

**Configuring for redistribution**

To configure for redistribution, define the redistribution tables with "restrict" redistribution filters. In the CLI, use the `restrict` command for RIP at the RIP router level.

**Syntax:**

```
restrict [ip-addr-mask | ip-addrpfx length]
```

This command prevents any routes with a destination address that is included in the range specified by the address/mask pair from being redistributed by RIP.

Do not enable redistribution until you have configured the redistribution filters. Otherwise, the network might become overloaded with routes that you did not intend to redistribute.

The default configuration permits redistribution for all default connected routes only.

**Example:**

To configure the switch to filter out redistribution of static, connected, or OSPF routes on network 10.0.0.0, enter the following commands:

```
switch(config)# router rip
switch(rip)# restrict 10.0.0.0 255.0.0.0
switch(rip)# write memory
```

The default configuration permits redistribution for all default connected routes only.
Modifying default metric for redistribution

The default metric is a global parameter that specifies the cost applied to all RIP routes by default. The default value is 1. You can assign a cost from 1 to 15.

Example:

To assign a default metric of 4 to all routes imported into RIP, enter the following commands:

```
switch(config)# router rip
switch(rip)# default-metric 4
```

Syntax:

default-metric value

The value can be from 1 to 15. The default is 1.

Enabling RIP route redistribution

The basic form of the redistribute command redistributes all routes of the selected type. For finer control over route selection and modification of route properties, you can specify the route-map parameter and the name of a route map. (For general information on route policy and route maps, see Route Policy. For examples of using route maps in route redistribution, see Using route policy in route redistribution.)

Do not enable redistribution until you have configured the redistribution filters. Otherwise, the network might become overloaded with routes that you did not intend to redistribute.

Syntax:

```
no router rip redistribute [connected | static | ospf] [route-map name]
```

Enables redistribution of the specified route type to the RIP domain.

static

Redistribute from manually configured routes.

connected

Redistribute from locally connected networks.

ospf

Redistribute from OSPF routes.

route-map name

Optionally specify the name of a route-map to apply during redistribution.

The no form of the command disables redistribution for the specified route type.

Example

To enable redistribution of all connected, static, and OSPF routes into RIP, enter the following commands.

```
switch(config)# router rip
switch(rip)# redistribute connected
switch(rip)# redistribute static
switch(rip)# redistribute ospf
switch(rip)# write memory
```

Changing the route loop prevention method
For more information about Poison reverse and Split horizon, see Changing the route loop prevention method.

**Syntax:**

```plaintext
no ip rip poison-reverse
```

Poison reverse is enabled by default. Disabling Poison reverse causes the routing switch to revert to Split horizon. (Poison reverse is an extension of Split horizon.) To disable Poison reverse on an interface, and thereby enable Split horizon, enter the following:

```plaintext
switch(config)# vlan 1
switch(vlan-1)# no ip rip poison-reverse
```

Entering the command without the `no` option re-enables Poison reverse.

### Viewing RIP information

All RIP configuration and status information is shown by the CLI command `show ip rip` and options off that command.

### Viewing general RIP information

**Syntax:**

```plaintext
show ip rip
```

To display general RIP information, enter `show ip rip` at any context level. The resulting display will appear similar to the following:

#### General RIP information listing

```plaintext
switch(config)# show ip rip

RIP global parameters
  RIP protocol : enabled  Auto-summary : enabled
  Default Metric : 4
  Distance : 120
  Route changes : 0
  Queries : 0

RIP interface information

<table>
<thead>
<tr>
<th>IP Address</th>
<th>Status</th>
<th>Send mode</th>
<th>Recv mode</th>
<th>Metric</th>
<th>Auth</th>
</tr>
</thead>
<tbody>
<tr>
<td>100.1.0.1</td>
<td>enabled</td>
<td>V2-only</td>
<td>V2-only</td>
<td>5</td>
<td>none</td>
</tr>
<tr>
<td>100.2.0.1</td>
<td>enabled</td>
<td>V2-only</td>
<td>V2-only</td>
<td>5</td>
<td>none</td>
</tr>
<tr>
<td>100.3.0.1</td>
<td>enabled</td>
<td>V2-only</td>
<td>V2-only</td>
<td>5</td>
<td>none</td>
</tr>
<tr>
<td>100.4.0.1</td>
<td>enabled</td>
<td>V2-only</td>
<td>V2-only</td>
<td>5</td>
<td>none</td>
</tr>
</tbody>
</table>

RIP peer information

<table>
<thead>
<tr>
<th>IP Address</th>
<th>Bad routes</th>
<th>Last update timeticks</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
```

The display is a summary of global RIP information, information about interfaces with RIP enabled, and information about RIP peers.

**RIP protocol**

Status of the RIP protocol on the router. RIP must be enabled here and on the VLAN interface for RIP to be active.

The default is disabled.

**Auto-summary**
Status of auto-summary for all interfaces running RIP. If auto-summary is enabled, subnets will be summarized to a class network when advertising outside of the given network.

**Default metric**
Sets the default metric for imported routes. This is the metric that will be advertised with the imported route to other RIP peers. A RIP metric is a measurement used to determine the "best" path to network: 1 is the best, 15 is the worst, 16 is unreachable.

**Route changes**
The number of times RIP has modified the routing switch’s routing table.

**Queries**
The number of RIP queries that have been received by the routing switch.

**RIP interface information**
RIP information on the VLAN interfaces on which RIP is enabled.

**IP address**
Address of the VLAN interface running RIP.

**Status**
Status of RIP on the VLAN interface.

**Send mode**
Format of the RIP updates: RIP 1, RIP 2, or RIP 2 version 1 compatible.

**Recv mode**
The switch can process RIP 1, RIP 2, or RIP 2 version 1 compatible update messages.

**Metric**
Path "cost," a measurement used to determine the "best" RIP route path: 1 is the best, 15 is the worst, 16 is unreachable.

**Auth**
RIP messages can be required to include an authentication key if enabled on the interface.

**RIP peer information**
RIP peers are neighboring routers from which the routing switch has received RIP updates:

**IP address**
IP address of the RIP neighbor.

**Bad routes**
Number of route entries which were not processed for any reason.

**Last update timeticks**
Number of seconds that have passed since we received an update from this neighbor.

**Viewing RIP interface information**
To display RIP interface information, enter the `show ip rip interface` command at any context level.

**Syntax:**

```
show ip rip interface [ip-addr | vlan vlan-id]
```
The resulting display will appear similar to the following:

```
switch(config)# show ip rip interface

RIP interface information

<table>
<thead>
<tr>
<th>IP Address</th>
<th>Status</th>
<th>Send mode</th>
<th>Recv mode</th>
<th>Metric</th>
<th>Auth</th>
</tr>
</thead>
<tbody>
<tr>
<td>100.1.0.1</td>
<td>enabled</td>
<td>V2-only</td>
<td>V2-only</td>
<td>1</td>
<td>none</td>
</tr>
<tr>
<td>100.2.0.1</td>
<td>enabled</td>
<td>V2-only</td>
<td>V2-only</td>
<td>1</td>
<td>none</td>
</tr>
<tr>
<td>100.3.0.1</td>
<td>enabled</td>
<td>V2-only</td>
<td>V2-only</td>
<td>1</td>
<td>none</td>
</tr>
<tr>
<td>100.4.0.1</td>
<td>enabled</td>
<td>V2-only</td>
<td>V2-only</td>
<td>1</td>
<td>none</td>
</tr>
</tbody>
</table>
```

You can also display the information for a single RIP VLAN interface, by specifying the VLAN ID for the interface, or by specifying the IP address for the interface.

**RIP interface output by VLAN**

To show the RIP interface information for VLAN 1000, use the `show ip rip interface vlan vid` command.

```
switch# show ip rip interface vlan 4

RIP configuration and statistics for VLAN 4

RIP interface information for 100.4.0.1

  IP Address : 100.4.0.1
  Status      : enabled
  Send Mode   : V2-only
  Recv mode   : V2-only
  Metric : 1
  Auth : none

  Bad packets received : 0
  Bad routes received  : 0
  Sent updates         : 0
```

For definitions of the fields in, see [Viewing general RIP information](#).

The RIP interface information also includes the following fields:

- **Bad packets received**
- **Number of packets that were received on this interface and were not processed for any reason.**
- **Bad routes received**
- **Number of route entries that were received on this interface and were not processed for any reason.**
- **Sent updates**
- **Number of RIP routing updates that have been sent on this interface.**

**Example:**

**Example of show IP rip interface output by IP address**

To show the RIP interface information for the interface with IP address 100.2.0.1, enter the `show ip rip interface` command:

```
switch# show ip rip interface 100.2.0.1

RIP interface information for 100.2.0.1

  IP Address : 100.2.0.1
```
Viewing RIP peer information

To display RIP peer information, enter the `show ip rip peer` command at any context level.

The resulting display will appear similar to the following:

**Example of show IP rip peer output**

```
switch# show ip rip peer
RIP peer information
IP Address      Bad routes  Last update timeticks
--------------- ------------ -------------------
100.1.0.100     0            1
100.2.0.100     0            0
100.3.0.100     0            2
100.10.0.100    0            1
```

This display lists all neighboring routers from which the routing switch has received RIP updates. The following fields are displayed:

**IP address**

IP address of the RIP peer neighbor.

**Bad routes**

The number of route entries that were not processed for any reason.

**Last update timeticks**

How many seconds have passed since the routing switch received an update from this peer neighbor.

To show the RIP peer information for a specific peer with IP address 100.1.0.100, enter

```
show ip rip peer 100.1.0.100
```

**Example of show IP rip peer ip-addr output**

```
switch# show ip rip peer 100.0.1.100
RIP peer information for 100.0.1.100
  IP Address : 100.1.0.100
  Bad routes : 0
  Last update timeticks : 2
```

This display lists information in the fields described above (IP address, Bad routes, Last update timeticks.)

Viewing RIP redistribution information

To display RIP redistribution information, enter the `show ip rip redistribute` command at any context level:

**Example of show IP rip redistribute output**
RIP automatically redistributes connected routes that are configured on interfaces that are running RIP and all routes that are learned via RIP. The `router rip redistribute` command (Configuring for redistribution), configures the routing switch to cause RIP to advertise connected routes that are not running RIP, static routes, and OSPF routes. The display shows whether RIP redistribution is enabled or disabled for connected, static, and OSPF routes.

### Viewing RIP redistribution filter (restrict) information

To display RIP restrict filter information, enter the `show ip rip restrict` command at any context level:

**Example of show IP rip restrict output**

```plaintext
switch# show ip rip restrict
RIP restrict list

<table>
<thead>
<tr>
<th>IP Address</th>
<th>Mask</th>
</tr>
</thead>
<tbody>
<tr>
<td>192.0.2.0</td>
<td>255.255.255.0</td>
</tr>
</tbody>
</table>
```

The display shows if any routes identified by the IP Address and Mask fields are being restricted from redistribution. The restrict filters are configured by the `router rip restrict` command. (See Configuring for redistribution.)

### RIP parameters and defaults

The following tables list the RIP parameters, their default values, and where to find configuration information.

#### RIP global parameters

The following table lists the global RIP parameters and their default values.

**Table 14: RIP global parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
<th>Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>RIP state</td>
<td>Routing Information Protocol V2-only.</td>
<td>Disabled</td>
</tr>
<tr>
<td>auto-summary</td>
<td>Enable/disable advertisement of summarized routes.</td>
<td>Enabled</td>
</tr>
<tr>
<td>metric</td>
<td>Default metric for imported routes.</td>
<td>1</td>
</tr>
<tr>
<td>redistribution</td>
<td>RIP can redistribute static, connected, and OSPF routes. (RIP redistributes connected routes by default, when RIP is enabled.)</td>
<td>Disabled</td>
</tr>
</tbody>
</table>
RIP interface parameters

The following table lists the VLAN interface RIP parameters and their default values.

Table 15: RIP interface parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
<th>Default</th>
</tr>
</thead>
</table>
| RIP version     | The version of the protocol that is supported on the interface. The version can be one of the following:  
|                 | - Version 1 only                                                                                 | V2-only        |
|                 | - Version 2 only                                                                                 |                |
|                 | - Version 1 or Version 2                                                                         |                |
| metric          | A numeric cost the routing switch adds to RIP routes learned on the interface. This parameter applies only to RIP routes. | 1             |
| IP address      | The routes that a routing switch learns or advertises can be controlled.                          | The routing switch learns and advertises all RIP routes on all RIP interfaces |
| loop prevention | The method the routing switch uses to prevent routing loops caused by advertising a route on the same interface as the one on which the routing switch learned the route:  
|                 | - Split horizon                                                                                  | Poison reverse |
|                 |   - The routing switch does not advertise a route on the same interface as the one on which the routing switch learned the route.  
|                 | - Poison reverse                                                                                 |                |
|                 |   - The routing switch assigns a cost of 16 "infinite" or "unreachable") to a route before advertising it on the same interface as the one on which the routing switch learned the route. |                |
| receive         | Define the RIP version for incoming packets                                                       | V2-only        |
| send            | Define the RIP version for outgoing packets                                                       | V2-only        |

Configuring RIP redistribution
You can configure the routing switch to redistribute connected, static, and OSPF routes into RIP. When you redistribute a route into RIP, the routing switch can use RIP to advertise the route to its RIP neighbors.
To configure redistribution, perform the following tasks:

1. Configure redistribution filters to permit or deny redistribution for a route based on the destination network address or interface. (optional)
2. Enable redistribution.

**Defining RIP redistribution filters**
Route redistribution imports and translates different protocol routes into a specified protocol type. On the switches, redistribution is supported for static routes, directly connected routes, and OSPF routes. Redistribution of any other routing protocol into RIP is not currently supported. When you configure redistribution for RIP, you can specify that static, connected, or OSPF routes are imported into RIP routes. Likewise, OSPF redistribution supports the import of static, connected, or RIP routes into OSPF routes.

**Changing the route loop prevention method**
For more information about Poison reverse and Split horizon, see Changing the route loop prevention method.

**Syntax:**

```
no ip rip poison-reverse
```

Poison reverse is enabled by default. Disabling Poison reverse causes the routing switch to revert to Split horizon. (Poison reverse is an extension of Split horizon.) To disable Poison reverse on an interface, and thereby enable Split horizon, enter the following:

```
switch(config)# vlan 1
switch(vlan-1)# no ip rip poison-reverse
```

Entering the command without the `no` option re-enables Poison reverse.
RIPv2 MD5 authentication

Introduction
Routing Information Protocol version 2 includes authentication types simple, none and MD5 (MD5 message-digest algorithm.) Both simple and none authentication types are vulnerable to passive attacks currently widespread in the Internet. Clear text passwords, currently specified for use with Routing Internet Protocol version 2 (RIPv2), are no longer considered sufficient to provide security. Keyed MD5 is the standard authentication algorithm for RIPv2. It provides a greatly enhanced probability that a system being attacked will detect and ignore hostile messages.

MD5 use case diagram

Configuration commands
Enable IP directed broadcast forwarding for Wake-on-LAN support. An optional ACL can also be applied to control what packets are forwarded.

Syntax
switch(config)# ip directed-broadcast [access-group <ACL-ID>]
access-group
Apply the specified access control list.
access-list-name-str
ASCII string specifying an ACL

Example configuration
switch(config)# ip directed-broadcast [access-group] <wol-acl>
Example running configuration

hostname "switch-name"
module 1 type j9573x
ip access-list extended "wol-acl"
....10 permit ip 192.168.1.10 0.0.0.0 182.168.1.1 0.0.0.255
....exit

ip directed-broadcast access-group "wol-acl"
ip routing
snmp-server community "public" unrestricted
ccbm
....ip address dhcp-bootp
exit
vlan 1
....name "DEFAULT_VLAN"
....no untagged 1,23-24
....untagged 2-22,25-26
....ip address dhcp-bootp
....exit
vlan 10
....name "VLAN10"
....untagged 1
....ip address 192.168.1.1 255.255.255.0
....exit
vlan 20
....name "VLAN20"
....untagged 23-24
....ip address 182.168.1.1 255.255.255.0
....exit

Figure 8  Configuration diagram
If specified ACL ID is non-existing, it is not possible to associate with IP Directed Broadcast. An error will be shown to the user.

It is not allowed to delete an ACL which is associated with IP Directed Broadcast and on attempt, an error message will be shown to user.

The same ACL `wol-acl` can be applied to any other interface like VLAN, port and tunnel.

**<wol-acl> entries**

```
ip access-list extended <wol-acl>
10 permit ip 192.168.1.1 255.255.255.0 182.168.1.1 55.255.255.0
20 deny ip 172.168.1.1 255.255.255.0 162.168.1.1 255.255.255.0
Exit
```

**Show commands**

IP directed broadcast hit counts for the associated access-list with can be displayed using the `show` command.

**Operating notes**

**Error messages**

**Configuring MD5 authentication for RIPv1 or RIPv1-or-v2 interface**

```
switch(vlan-10)# ip rip v1-only
switch(vlan-10)# ip rip authentication-type md5
```

Only RIPv2 interfaces support MD5 authentication.

```
switch(vlan-10)# ip rip v1-or-v2
switch(vlan-10)# ip rip authentication-type md5
```

Only RIPv2 interfaces support MD5 authentication

**Configuring MD5 keychain for RIPv2 interface without keychain or key or keystring**

```
switch(vlan-10)# ip rip md5-auth-key-chain rip-md5-chain
Chain rip-md5-chain is not found.
```

```
switch(vlan-10)# ip rip md5-auth-key-chain rip-md5-chain
Chain rip-md5-chain has no keys configured.
```

```
switch(vlan-10)# ip rip md5-auth-key-chain rip-md5-chain
Chain rip-md5-chain has no keystring configured.
```
**RIPng for IPv6**

While the mechanisms of RIP remain unchanged, RIPng for IPv6 has been added to include support for IPv6 addressing and prefixes, different packet formats, packet lengths and no authentication on switches. RIPng is for IPv6 only just as RIPv2 is for IPv4 only. RIPv2 and RIPng must be regarded as two independent protocols with no interaction between them. RIPng is specified by RFC 2080 and RFC 2081

---

**NOTE**

RIPng and RIPv2 can be supported on the same interface/VLAN.

---

**Supported features**

- RIPng global enable/disable Enables/Disables RIPng protocol in the config context.
- Split horizon Prevents the formation of loops in routing. A router is not allowed the advertisement of routes back to the interface where it was initially learned. Enabled by default. Split Horizon is a non-configurable feature.
- Poison-Reverse Optimizes the transmission of routing information and improves the time-to-reach network convergence. Enabled by default and can be disabled per VLAN interface.
- Redistribute connected/static/ospfv3 routes RIPng protocol advertises routes learned from static, connected and other routing protocols (example OSPFv3) to its peers.
- Metric configuration for imported routes Updates the metric for imported routes based on the value configured.
  - router ripng default-metric — for routes imported from protocols other than RIPng
  - vlan <id> ipv6 ripng metric — for routes received from other RIPng peer
- Configuration of RIPng timers: update, timeout and garbage collect.
  - Update timer defines interval between update messages.
  - Timeout timer defines route aging time.
  - The garbage-collect timer defines the time interval when the metric of a route is 16 to the time when it is deleted from the routing table.
- Administrative distances: The default value can be modified and the value is applied to all routes learned through RIPng.
- RIPng will listen only to RIPng packets sent to the multicast address FF02::9. All packets sent out will be addressed to FF02::9 and the source IP will be the linklocal IP address of the VLAN.
- Route maps—Route maps are applied in the redistribution process to control route prefixes or to modify the attributes of the routes. Route-maps can be used in RIPng redistribution to apply route policy configurations.
- RIPng notifications/traps—Traps are generated as the result of finding an unusual condition while parsing an RIPng packet or a processing a timer event. Disabled by default.

---

**Limitations**

Limits imposed on RIPng are as follows:
IPv6 loop back addresses cannot be redistributed into RIPng

<table>
<thead>
<tr>
<th>Type of limit</th>
<th>Maximum limit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of interfaces/VLANs on which RIPng may be run:</td>
<td>128</td>
</tr>
<tr>
<td>Total number of routes supported:</td>
<td>5000</td>
</tr>
<tr>
<td>Maximum number of IPv6 addresses per VLAN:</td>
<td>32</td>
</tr>
<tr>
<td>Maximum number of IPv6 VLANs:</td>
<td>256 512</td>
</tr>
<tr>
<td>Maximum number of IPv6 addresses:</td>
<td>2048</td>
</tr>
</tbody>
</table>

**Configure RIPng**

From within the configuration context, use the following commands to configure, enable, disable a RIPng setting.

**Enable/Disable RIPng global**

**Syntax**

`router ripng enable | disable`

**Description**

From within the configuration context, enable RIPng globally or disable RIPng globally.

**Configure a RIPng setting**

**Syntax**

`router ripng`<br>`no router ripng`

**Description**

From within the configuration context, configure a RIPng setting or enter RIPng context. Use the `no` argument to remove all RIPng configurations.

**Configure a default metric**

**Syntax**

`router ripng default-metric 1-15`

**Description**

Configure a default metric for routes that are imported from protocols other than RIPng. The default value is 1.

**Configure the administrative distance for routes**

**Syntax**

`router ripng distance 1-255`

**Description**

Configure the administrative distance for routes that are learned via RIPng. The default value is 120.

**Redistribute router RIPng**
**Syntax**
```
router ripng redistribute
```

**Description**
Redistribute connected/static/other protocols routes.
Use `no` to disable redistribution of the specified protocol.

**Options**
- **connected**
  Redistribute locally connected networks.

- **ospf3**
  Redistribute OSPFv3 routes.

- **static**
  Redistribute manually configured routes.

- **include-all**
  Include null and reject routes.
  
  Include-all option is only for static routes.

- **route-map**
  Redistribute a route map.
  
  Route-map option comes only after we specify the protocol (static/connected/ospf3).

**Usage**
```
no redistribute connected route-map NAME
no redistribute ospf3 route-map NAME
no redistribute static include-all route-map NAME
```

**Configure RIPng timers**

**Syntax**
```
router ripng timers
```

**Description**
Configure RIPng timers.

**Options**
- **garbage-collect**
  Set the garbage-collect interval for the route.
  The default value is 120 seconds.

- **timeout**
Set the interval for the route timeout.  
The default value is 180 seconds.

**update**  
Set the interval for the update timer.  
The default value is 30 seconds.

**NOTE**  
It is recommended not to change the default values.

**Usage**  
router ripng timers garbage-collect 5-65535  
router ripng timers timeout 5-65535  
router ripng timers update 5-65535

**Enable/Disable RIPng traps**

**Syntax**  
router ripng trap

**Description**  
Enable/Disable RIPng traps.

**Options**  
Traps are generated as the result of finding an unusual condition while parsing an RIPng packet or a processing a timer event. If more than one type of unusual condition is encountered while parsing the packet or processing an event, only the first one will generate a trap.

- interface-state-change
- Send a trap when the state of an interface changes.
- interface-config-error
- Send a trap when a configuration conflict occurs for an interface.
- interface-receive-bad-packet
- Send a trap when an invalid packet is received on an interface.
- all

Enable all the RIPng traps.

**Usage**  
trap TRAP-NAME | all  
no trap TRAP-NAME | all

**VLAN Level Configuration**

This is a VLAN context command. It can be entered in VLAN context as shown or following the

- vlan VLAN-ID
- command.

**IPv6 RIPng**

**Syntax**  
ipv6 ripng  
no ipv6 ripng

**Description**  
Enables/disables/configures the RIPng protocol for IPv6 on the interface.
The argument no disables or disconfigures RIPng on the interface.

**Options**

**enable**
Enable RIPng on the VLAN.

**metric**
Set the metric for the interface.

**poison-reverse**
Enable/Disable poison reverse.

**Show commands**

IP directed broadcast hit counts for the associated access-list with can be displayed using the `show` command.

**Show IPv6 ripng general**

**Syntax**

```plaintext
show ipv6 ripng general
```

**Description**

Displays RIPng global parameters only as shown below.

**RIPng global parameters**

```plaintext
switch(config)# show ipv6 ripng general
RIPng global parameters
RIPng protocol : Enabled
Default metric : 1
Administrative distance : 120
Route changes : 1090
Queries : 134457
Update time : 30
Timeout : 180
Garbage-collect time : 120
HP-3810M-24GT-1s(config)#
```

**Show IPv6 ripng interface**

**Syntax**

```plaintext
show ipv6 ripng interface
```

**Description**

Displays basic config, interface and peer information as shown below.

**Options**

**VLAN**

Specify the VLAN of the interface requesting detailed information.

**VLAN-ID**

Enter a VLAN identifier or a VLAN name.

**Usage**

```plaintext
show ipv6 ripng interface vlan VLAN-ID
```
RIPng interface information

```
switch(config)# show ipv6 ripng
RIPng global parameters
RIPng protocol : Enabled
Default metric : 1
Administrative distance : 120
Route changes : 2090
Queries : 134877
Update time : 30
Timeout : 180
Garbage-collect time : 120

RIPng interface information
VLAN Status Metric
----------- ----------- -----------
10 Enabled 1
20 Enabled 1

RIPng peer information
IPv6 Address Bad Last update
packets timeticks
---------------------------------- ---- -----
fe80::200:eff:feda:98b6%vlan10 0 27
```

Show IPv6 RIPng peer

**Syntax**

```
show ipv6 ripng peer
```

**Description**

Shows the peers learned through RIPng.

**RIPng peer information**

```
switch (config)# show ipv6 ripng peer
RIPng peer information
IPv6 Address Bad packets Last update
packets timeticks
---------------------------------- ---- -----
fe80::ab23:ccff:fef4:fc40 0 30
```

Since RIPng does not have an active peering mechanism, this command shows only those RIPng peers from which a route was taken and added to the routing table. For example, if two peers advertise the same route(s) with the same metric only one of them will be shown as peer.

Show IPv6 RIPng redistribute

**Syntax**

```
show ipv6 ripng redistribute
```

**Description**

List the protocols that are being redistributed into RIPng.

**RIPng redistributing without route-maps**
switch(config)#show ipv6 ripng redistribute
RIPng redistributing
Route type Route map Options
----------------- ------------------ -------------------------------
Connected

**RIPng redistribute with route-maps**

switch(config)#show ipv6 ripng redistribute
RIPng redistributing
<table>
<thead>
<tr>
<th>Route type</th>
<th>Route map</th>
<th>Options</th>
</tr>
</thead>
<tbody>
<tr>
<td>Connected</td>
<td>map2</td>
<td></td>
</tr>
<tr>
<td>static</td>
<td>map1</td>
<td>Include null-route and reject</td>
</tr>
</tbody>
</table>

**Show IPv6 RIPng traps**

**Syntax**

show ipv6 ripng traps

**Description**

Display the enabled RIPng traps.

**RIPng Traps : Enabled**

switch(config)#show ipv6 ripng traps
RIPng Traps : Enabled
  RIPng Traps Enabled
  -------------------
  Interface State Change
  Interface Configuration Error
  Interface Bad Packet Receive Error

**Show IPv6 route RIPng**

**Syntax**

show ipv6 route ripng

**Description**

Show the IPv6 routing table. The output can be restricted to a specific destination or type of route.

**Options**

IPv6-ADDR

The destination IPv6 address for which to display the routes.

**Usage**

show ipv6 route IPv6-ADDR static | connected | ospfv3 | ripng

**IPv6 route entries**

switch# show ipv6 route
IPv6 Route Entries
T (Type):
S: Static  C: Connected

<table>
<thead>
<tr>
<th>Destination/ Metric</th>
<th>Gateway</th>
<th>T</th>
<th>ST</th>
<th>Distance</th>
</tr>
</thead>
<tbody>
<tr>
<td>--</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Show ipv6 route summary

**Syntax**

show ipv6 route summary

**Description**

Show the summary of IPv6 routing table.

### IPv6 route summary

```
switch (config)# show ipv6 route summary
IPv6 Route Table Summary
Protocol Active Routes
-----------------------
Connected 5
Ripng 4000
```

**Debug commands**

### Debug IPv6 RIPng

**Syntax**

dump ipv6 ripng

**Description**

Enable debug messages for RIPng.

**Options**

database

Show RIPng database changes.

events

Show RIPng events.

trigger

Show RIPng trigger messages.

**Usage**

dump ipv6 ripng database | events | trigger

**Additional commands**

Following CLI commands are enhanced to accommodate RIPng.

### VLAN VLAN-ID IPv6

This is a VLAN context command.

**Syntax**

```
vlan VLAN-ID ipv6 ripng
```

**Description**

Enables/Disables/Configures RIPng protocol for IPv6 on the interface. The command `no ipv6 ripng enable` disables or disconfigures RIPng on the interface. This command can be followed by a RIPng configuration command.

### Show running config

**Syntax**

```
show running-config router {rip | bgp | ospf | ospf3 | vrrp | ripng}
```

**Description**

Show the running configuration for layer 3 routing protocols.

**Options**

- `show running-config router bgp`
  
  Show the running configuration for bgp.

- `show running-config router ospf`
  
  Show the running configuration for ospf.

- `show running-config router ospf3`
  
  Show the running configuration for OSPFv3.

- `show running-config router rip`
  
  Show the running configuration for RIP.

- `show running-config router ripng`
  
  Show the running configuration for RIPng.

- `show running-config router vrrp`
  
  Show the running configuration for VRRP.

### Show running-config router ripng

```
switch (config)# show running-config router ripng
router ripng
enable
default-metric 3
distance 95
redistribute connected
redistribute static
redistribute ospf3
```

### Show running-config vlan

**Syntax**

```
show running-config vlan VLAN-ID
```

**Description**

Shows the IPv6 ripng vlan configuration along with other vlan specific configuration.
<table>
<thead>
<tr>
<th>Command</th>
</tr>
</thead>
<tbody>
<tr>
<td>ipv6 address 3005::10/64</td>
</tr>
<tr>
<td>ipv6 ripng enable</td>
</tr>
</tbody>
</table>
Creating a Policy Based Routing Policy

Policy Based Routing (PBR) enables you to manipulate a packet’s path based on attributes of the packet. Traffic with the same destination can be routed over different paths, so that different types of traffic, such as VOIP or traffic with special security requirements, can be better managed.

PBR is available on switches running v2 or higher modules.

The supported actions for PBR are:

- Setting the next hop for routing the packet if there is no explicit route for this destination (`[ipv4 | ipv6] ip default-next-hop [ip-addr]`).
- Setting the outbound tunnel interface for the packet (`interface tunnel [tunnel-ID]`). See the IPv6 Configuration Guide for your switch.
- Setting `interface null-route`, which specifies that the packets are dropped if no other actions have occurred.

Operating notes for PBR

- Multiple actions can be configured for a class, up to 8 actions per class.
- If you configure an action of interface null-route, no more actions for that class may be configured.
- Only one of the 8 possible actions can be active at one time.
- The precedence of actions is indicated by the order in which they are added to the policy.
- Actions can only be added to a class, and they are added to the end of the action list for the class.
- To remove actions from a class, the entire class must be removed from the policy.
- When an action becomes inactive, for example, if the configured address becomes unreachable (for next-hop and default-next-hop) or the interface goes down (for a tunnel), the policy is configured with the next action for that class, if possible. If that action is not active, the next action is tried, and so on, until an interface null-route or the end of the list of configured actions is encountered. If the end of the list is reached, the policy action for that class behaves as if no PBR policy is applied.
- The maximum combined number of unique IP next-hops and default-next-hops supported is 16.
- PBR can be applied only to unicast traffic.

TCP and UDP traffic routing

The following example shows TCP and UDP traffic routed on different network paths. First, the traffic classes are created, then the PBR policy is created, and lastly the PBR policy is applied to an interface.

```
switch(config)# class ipv4 TCP
switch(config-class)# match tcp 10.0.8.1/24 15.29.16.104/24 eq 80
switch(config-class)# match tcp 10.0.8.1/24 15.29.16.104/24 eq 22
switch(config-class)# match tcp 10.0.8.1/24 15.29.16.104/24 eq 23
switch(config-class)# exit
```
To enable debug logging for PBR, enter the debug ip pbr command. A message is logged when a PBR policy is applied, when the action in a class becomes inactive, and when an action in a class becomes active. See the Management and Configuration Guide for your switch.

---

**NOTE**

PBR is available on switches running v2 or higher modules.

## Troubleshooting PBR

### Cause

Use the show statistics policy command to display information about which PBR action for an applied policy is active. Hit counts for each entry in the class and policy with the active action are displayed.

```plaintext
switch(vlan-111)# show statistics policy TCP_UDP vlan 111 in
HitCounts for Policy TCP_UDP
Total
100 class ipv4 TCP action
( 5 ) 10 match tcp 10.0.8.1 0.0.0.255 15.29.16.104 0.0.0.255 eq 80
( 2 ) 20 match tcp 10.0.8.1 0.0.0.255 15.29.16.104 0.0.0.255 eq 22
( 2 ) 30 match tcp 10.0.8.1 0.0.0.255 15.29.16.104 0.0.0.255 eq 23
110 class ipv4 voice action
( 4 ) 10 match tcp 10.0.8.1 0.0.0.255 15.29.16.104 0.0.0.255 eq 80
```

To enable debug logging for PBR, enter the debug ip pbr command. A message will be logged when a PBR policy is applied, when the action in a class becomes inactive, and when an action in a class becomes active. See the Management and Configuration Guide for your switch.
OSPFv2 is the IPv4 implementation of the Open Shortest Path First protocol. (OSPFv3 is the IPv6 implementation of this protocol.) Beginning with software version K.15.01, the switches can be configured to run OSPFv2 either alone or simultaneously with OSPFv3. (OSPFv2 and OSPFv3 run as independent protocols on the switch and do not have any interaction when run simultaneously.)

For overview information on OSPF, see Overview of OSPF.

**Configuring OSPF on the routing switch**

**Enabling IP routing**

**Syntax:**
```
no ip routing
```

Executed at the global configuration level to enable IP routing on the routing switch.

**Default:** Disabled

The no form of the command disables IP routing. (Global OSPF and RIP routing must be disabled before you disable IP routing.)

**Example:**
```
switch(config)# ip routing
```

**Enabling global OSPF routing**

**Syntax:**
```
router ospf [enable | disable]
```

Executed at the global configuration level to enable OSPF on the routing switch and to enter the OSPF router context. This enables you to proceed with assigning OSPF areas, including area border router (ABR) and autonomous system boundary router (ASBR) configuration, and to modify OSPF global parameter settings as needed.

The **enable** form of the command enables OSPF routing, and the **disable** form of the command disables OSPF routing.

Global IP routing must be enabled before executing this command.

**Default:** Disabled

The no form of the command deletes all protocol specific information from the global context and interface context. All protocol parameters are set to default values.

---

**NOTE**

If you disable OSPF, the switch retains all the configuration information for the disabled protocol in flash memory. If you subsequently restart OSPF, the existing configuration will be applied. After restarting OSPF, the exiting configuration will be applied and the protocol will be in the disabled state.

---

**Example**
To enter the OSPF router context

switch(config)#router ospf
switch(ospf)#

To enable OSPF routing

switch(config)#router ospf enable
switch(ospf)#

To disable OSPF routing

switch(config)#router ospf disable
switch(ospf)#

Note: The no router ospf enable command also disables OSPF routing.

To delete all protocol-specific information from the global context and interface context and set all protocol parameters to default values:

switch(config)#no router ospf
switch(ospf)#

Changing the RFC 1583 OSPF compliance setting

For more information, see Changing the RFC 1583 OSPF compliance setting.

Syntax:

rfc1583-compatibility

Executed at the global configuration level to toggle routing switch operation compliance between RFC 1583 and RFC 2328.

rfc1583-compatibility

Configures the routing switch for external route preference rules compliant with RFC 1583.

no rfc1583-compatibility

Configures the routing switch for external route preference rules compliant with RFC 2328.

Default: Compliance enabled

Example

To disable RFC 1583 compatibility on a routing switch in an OSPF domain where RFC 2178 and RFC 2328 are universally supported:

switch(config)# router ospf
switch(ospf)# no rfc1583-compatibility

Changing external route preference compatibility from RFC 1583 to RFC 2328
Assigning the routing switch to OSPF areas

For more information, see Assigning the routing switch to OSPF areas.

Configuring an OSPF backbone or normal area

Syntax:
```
area [(ospf-area-id) | [backbone]) [normal] [(ospf-area-id) | [backbone])
```

After using `router ospf` to globally enable OSPF and enter the global OSPF context, execute this command to assign the routing switch to a backbone or other normal area.

The no form of the command removes the routing switch from the specified area.

Default: No areas; Range: 1 to 16 areas (of all types)

ospf-area-id

Specifies a normal area to which you are assigning the routing switch. You can assign the routing switch to one or more areas, depending on the area in which you want each configured VLAN or subnet to reside.

You can enter area IDs in either whole number or dotted decimal format. (The routing switch automatically converts whole numbers to the dotted decimal format.)

For example, if you enter an area-ID of 1, it appears in the switch's configuration as 0.0.0.1 and an area-ID of 256 appears in the switch configuration as 0.0.1.0.

An area ID can be a value selected to match the IP address of a VLAN belonging to the area or a value corresponding to a numbering system you devise for the areas in a given autonomous system (AS.)

Entering an area ID of 0 or 0.0.0.0 automatically joins the routing switch to the backbone area.

The maximum area ID value is 255.255.255.254 (4,294,967,294.)

backbone

Assigns the routing switch to the backbone area and automatically assigns an area ID of 0.0.0.0 and an area type of normal.

Using 0 or 0.0.0.0 with the above ospf-area-id option achieves the same result. The backbone area is automatically configured as a normal area type.

Example

To configure a backbone and a normal area with an ID of "1" (0.0.0.1) on a routing switch:
Configuring a stub or NSSA area

**Syntax:**

```plaintext
area ospf-area-id stub 0-16777215 [no-summary] area ospf-area-id nssa 0-16777215 [no-summary] [metric-type [type1 | type2]] No area ospf-area-id
```

After using `router ospf` to globally enable OSPF and enter the global OSPF context, execute this command to assign the routing switch to a stub area or NSSA. (Does not apply to backbone and normal OSPF area ABRs.)

The `no` form of the command removes the routing switch from the specified area.

Default: No areas; Range: 1 to 16 areas (of all types)

**ospf-area-id**

Same area ID as in Configuring an OSPF backbone or normal area, except you cannot assign a backbone area number (0 or 0.0.0.0) to a stub or NSSA area.

**[stub | nssa]**

Designates the area identified by `ospf-area-id` as a stub area or NSSA.

0-16777215

If the routing switch is used as an ABR for the designated area, assigns the cost of the default route (to the backbone) that is injected into the area.

---

**NOTE**

If the routing switch is not an ABR for the stub area or NSSA, the above cost setting is still required by the CLI, but is not used.

---

In the default configuration, a routing switch acting as an ABR for a stub area or NSSA injects type-3 summary routes into the area. For an NSSA, the routing switch also injects a type-7 default route into the area.

**[no-summary]**

Where the routing switch is an ABR for a stub area or an NSSA, this option reduces the amount of link-state advertisement (LSA) traffic entering the area from the backbone by replacing the injection of type-3 summary routes with injection of a type-3 default summary route.

For NSSAs, this command also disables injection of the type-7 default external route from the backbone into the area (included in the metric-type operation described below.)

Default: Disabled

For more information, see Not-so-stubby-area (NSSA) on page 222 and Replacing type-3 summary LSAs and type-7 default external LSAs with a type-3 default route LSA on page 224.

**[metric-type [type1 | type2]]**

Used in NSSA ABRs only.

Enables injection of the type-7 default external route and type-3 summary routes into the area instead of a type 3 default route. Also specifies the type of internal cost metric to include in type-7 LSAs advertised for redistribution of external routes in the NSSA. (The redistribution—or external—cost metric is a global setting on the routing switch set by the `default-metric` command.)

The `metric-type` command specifies whether to include the redistribution cost in the cost metric calculation for a type-7 default LSA injected into the area.
Calculate external route cost for a type-7 default LSA as the sum of (1) the external route cost assigned by the ASBR plus (2) the internal cost from the router with traffic for the external route to the ASBR advertising the route.

If metric-type is not specified, the default (type2) will be used.

Using the `area ospf-area-id nssa 0-16777215` without entering either `no-summary` or `metric-type` resets the routing switch to the state where injection of type-3 summary routes and the type-7 default external routes is enabled with `metric-type set to type2`.

Default: Enabled with `metric-type type2`

**Examples**

The following examples of configuring a stub area and an NSSA on a routing switch use an (arbitrary) cost of "10".

**Creating stub area and NSSA assignments**

- `Switch(ospf)# area 2 stub 10` Assigns a stub area with a cost of 10.
- `Switch(ospf)# area 3 nssa 10` Assigns an NSSA with a cost of 10 and, by default, uses a Type2 default cost metric for Type-7 (external) route LSAs received from the backbone.
- `Switch(ospf)# area 4 nssa 10 no-summary` Assigns an NSSA with a cost of 10, blocks injection of type-3 summary routes, and starts injection of type-3 default routes from the backbone.
- `Switch(ospf)# area 5 nssa 10 metric-type type1` Sets the cost metric type for type-7 (default) LSAs injected into the NSSA.

**Assigning VLANs and/or subnets to each area**

After you define an OSPF area (page A-25), you can assign one or more VLANs and/or subnets to it. When a VLAN is assigned to an area, all currently configured IP addresses in that VLAN are automatically included in the assignment unless you enter a specific IP address.

All static VLANs configured on a routing switch configured for OSPF must be assigned to one of the defined areas in the AS.

**Syntax:**

```
vlan vid # ip ospf [ip-addr | all] area ospf-area-id
```

Executed in a specific VLAN context to assign the VLAN or individual subnets in the VLAN to the specified area. Requires that the area is already configured on the routing switch (page A-25.)

When executed without specifying an IP address or using the `all` keyword, this command assigns all configured networks in the VLAN to the specified OSPF area.
vlan vid
Defines the VLAN context for executing the area assignment.

ip-addr
Defines a specific subnet on the VLAN to assign to a configured OSPF area.

all
Assigns all subnets configured on the VLAN to a configured OSPF area.

area ospf-area-id
Identifies the OSPF area to which the VLAN or selected subnet should be assigned.

If you add a new subnet IP address to a VLAN after assigning the VLAN to an OSPF area, you must also assign the new subnet to an area:

- If all subnets in the VLAN should be assigned to the same area, just execute `ip ospf area ospf-area-id`
- But if different subnets belong in different areas, you must explicitly assign the new subnet to the desired area.

Also, to assign a VLAN to an OSPF area, the VLAN must be configured with at least one IP address. Otherwise, executing this command results in the following CLI message:

```
OSPF can not be configured on this VLAN.
```

Example
To assign VLAN 8 on a routing switch to area 3 and include all IP addresses configured in the VLAN, enter the following commands:

```
switch(ospf)# vlan 8
switch(vlan-8)# ip ospf area 3
```

Suppose that a system operator wants to assign the three subnets configured in VLAN 10 as shown below:

- 10.10.10.1 to OSPF area 5
- 10.10.11.1 to OSPF area 5
- 10.10.12.1 to OSPF area 6

The operator could use the following commands to configure the above assignments:

```
switch(ospf)# vlan 10
switch(vlan-10)# ip ospf 10.10.10.1 area 5
switch(vlan-10)# ip ospf 10.10.11.1 area 5
switch(vlan-10)# ip ospf 10.10.12.1 area 6
```

Assigning loopback addresses to an area
Optional: After you define the OSPF areas to which the switch belongs, you can assign a user-defined loopback address to an OSPF area. A loopback interface is a virtual interface configured with an IP address and is always reachable as long as at least one of the IP interfaces on the switch is operational. Because the loopback interface is always up, you ensure that the switch's router ID remains constant and that an OSPF network is protected from changes caused by downed interfaces.
For more information, see the management and configuration guide for your switch.

Syntax:
```
interface loopback 0-7 ip ospf lo-ipaddress area ospf-area-id
```
Executed in a specific loopback context to assign a loopback interface to the specified OSPF area. Requires that the specified loopback interface is already configured with an IP address on the switch.
```
interface loopback 0-7
```
Defines the loopback context for executing the area assignment.
```
ip ospf lo-ipaddress
```
Specifies the loopback interface by its IP address to assign to a configured OSPF area.
```
area ospf-area-id
```
Identifies the OSPF area to which the loopback interface is assigned.
You can enter a value for the OSPF area in the format of an IP address or a number in the range 0 to 4,294,967,295.

Example:
To assign user-defined loopback interface 3 on the switch to area 192.5.0.0 and include the loopback IP address 172.16.112.2 in the OSPF broadcast area, enter the following commands:
```
switch(config)# interface loopback 3
switch(lo-3)# ip ospf 172.16.112.2 area 192.5.0.0
```

Syntax:
```
interface loopback 0-7# ip ospf lo-ip-address cost number
```
Executed in a specific loopback context to modify the cost used to advertise the loopback address (and subnet) to the area border router (ABR.) Requires that the specified loopback interface is already configured with an IP address on the switch.
```
loopback interface 0-7
```
Defines the loopback context for executing the cost assignment.
```
ip ospf lo-ip-address
```
Specifies the loopback interface by its IP address.
```
cost number
```
Specifies a number that represents the administrative metric associated with the loopback interface. Valid values are from 1 to 65535.
Default: 1.

Example
To configure a cost of 10 for advertising the IP address 172.16.112.2 configured for loopback interface 3 in an OSPF area 192.5.0.0, enter the following commands:
```
switch(config)# interface loopback 3
switch(lo-3)# ip ospf 172.16.112.2 area 192.5.0.0
switch(lo-3)# ip ospf 172.16.112.2 cost 10
```

**OSPF redistribution of loopback addresses**

When you assign a loopback address to an OSPF area, the route redistribution of the loopback address is limited to the specified area.

When route redistribution is enabled:
The switch advertises a loopback IP address that is not assigned to an OSPF area as an OSPF external route to its OSPF neighbors, and handles it as a connected route.

The switch advertises a loopback address that is assigned to an OSPF area as an OSPF internal route.

To enable redistribution of loopback IP addresses in OSPF, enter the `redistribution connected` command as described in Enabling route redistribution.

Example

Assigning loopback IP addresses to OSPF areas

The loopback IP address 13.3.4.5 of loopback 2 is advertised only in OSPF area 0.0.0.111. The IP addresses 14.2.3.4 and 15.2.3.4 of loopback 1 are advertised in all OSPF areas. The lines in bold below show that the IP address of loopback interface 2 is assigned to OSPF area 111.

```
switch(config)# interface loopback 1
switch(lo-1)# ip address 14.2.3.4
switch(lo-1)# ip address 15.2.3.4
switch(lo-1)# exit
switch(config)# interface loopback 2
switch(lo-2)# ip address 13.3.4.5
switch(lo-2)# ip ospf 15.2.3.4 area 0.0.0.111
switch(lo-2)# exit
```

Verifying OSPF redistribution of loopback interfaces

To verify the OSPF redistribution of loopback interfaces, enter the `show ip route` command from any context level to display IP route table entries.

In this example, a loopback address assigned to an area is displayed as an `ospf intra-area (internal)` route to its neighbor; a loopback address not assigned to a specific area is displayed as an `ospf external` route:

```
switch(config)# show ip route

IP Route Entries
Destination    Gateway      VLAN Type Sub-Type Metric  Dist
------------    -------      ---- ---- ---- ---- ---- ----
20.0.15.1/32    25.0.67.131  25  ospf  external2 10   110
20.0.16.2/32    25.0.67.131  25  ospf  intra-area 2   110
```

Configuring external route redistribution in an OSPF domain (optional)

For more information, see Configuring for external route redistribution in an OSPF domain.

Configuring redistribution filters

Syntax:
```
router ospf restrict ip-addr/mask-length
```
Prevents distribution of the specified range of external routes through an ASBR from sources external to the OSPF domain. This will prevent external routes with the specified IP address/mask from entering the OSPF domain.

---

This command can be used to help implement inbound traffic filtering.

Default: Allow all supported, external route sources.

---

Use this command to block unwanted, external routes before enabling route redistribution on the ASBR.

Example:

To configure a routing switch operating as an ASBR to filter out redistribution of static, connected, or RIP routes on network 10.0.0.0, enter the following commands:

```
switch(config)# router ospf restrict 10.0.0.0/8
```

In the default configuration, redistribution is permitted for all routes from supported sources.

---

Enabling route redistribution

This step enables ASBR operation on a routing switch, and must be executed on each routing switch connected to external routes you want to redistribute in your OSPF domain.

The basic form of the `redistribute` command redistributes all routes of the selected type. For finer control over route selection and modification of route properties, you can specify the `route-map` parameter and the name of a route map. (For general information on route policy and route maps, see Route Policy. For examples of using route maps in route redistribution, see Using route policy in route redistribution.)

---

Do not enable redistribution until you have configured the redistribution "restrict" filters. Otherwise, the network might become overloaded with routes that you did not intend to redistribute.

---

Syntax:

```
router ospf redistribute [connected | static | rip] route-map name
```

Executed on an ASBR to globally enable redistribution of the specified route type to the OSPF domain through the area in which the ASBR resides.

- **static**
  - Redistribute from manually configured routes.
- **connected**
  - Redistribute from locally connected networks.
- **rip**
  - Redistribute from RIP routes.

Optionally specify the name of a route-map to apply during redistribution.

The `no` form of the command disables redistribution for the specified route type.

Example

To enable redistribution of all supported external route types through a given ASBR, execute the following commands.
### Modifying the default metric for redistribution

Optional: The default metric is a global parameter that specifies the cost applied to all OSPF routes by default.

**Syntax:**
```
router ospf default-metric 0-16777215
```

Globally assigns the cost metric to apply to all external routes redistributed by the ASBR. By using different cost metrics for different ASBRs, you can prioritize the ASBRs in your AS.

Default: 10

**Example:**
To assign a default metric of 4 to all routes imported into OSPF on an ASBR, enter the following commands:
```
switch()#
switch(config)# router ospf default-metric 4
```

### Modifying the redistribution metric type

Optional: The redistribution metric type is used by default for all routes imported into OSPF. Type 1 metrics are the same "units" as internal OSPF metrics and can be compared directly. Type 2 metrics are not directly comparable, and are treated as larger than the largest internal OSPF metric.

**Syntax:**
```
router ospf metric-type [type1 | type2]
```

Globally reconfigures the redistribution metric type on an ASBR.

- **type1**
  - Specifies the OSPF metric plus the external metric for an external route.
- **type2**
  - Specifies the external metric for an external route.

Default: type2

**Example:**
To change from the default setting on an ASBR to type 1, enter the following command:
```
switch(config)# router ospf metric-type type1
```

### Configuring ranges on an ABR to reduce advertising to the backbone

**Syntax:**
```
area [ospf-area-id | backbone] range [ip-addr/mask-length] [no-advertise] [type summary | nssa | cost 1-16777215]
```

Use this command on a routing switch intended to operate as an ABR for the specified area to do either of the following:

- Simultaneously create the area and corresponding range setting for routes to summarize or block.
- For an existing area, specify a range setting for routes to summarize or block.
Options

ospf-area-id

Same area ID as in Configuring an OSPF backbone or normal area, except you cannot assign a backbone area number (0 or 0.0.0.0) to a stub or NSSA area.

range ip-addr/mask-length

Defines the range of route advertisements to either summarize for injection into the backbone area or to prevent from being injected into the backbone area. The ip-addr value specifies the IP address portion of the range, and mask-length specifies the leftmost significant bits in the address. The ABR for the specified area compares the IP address of each outbound route advertisement with the address and significant bits in the mask to determine which routes to select for either summarizing or blocking. For example, a range of 10.10.32.1/14 specifies all routes in the range of 10.10.32.1 - 10.10.35.254.

no advertise

Use this keyword only if you want to configure the ABR to prevent advertisement to the backbone of a specified range of routes. (This has the effect of "hiding" the specified range from the backbone area.) If you do not use this option, the ABR advertises the specified range of routes according to the

type summary | nssa

selection described below.

[type summary [{cost 1-16777215] | nssa}]

Configures the type of route summaries to advertise or block.

If the option type is not used in the command, the ABR defaults this setting to summary.

The option type summary [cost 1-16777215] specifies internal routes in the configured range of RAs and the user-configured cost for an area summary range. If cost is specified, the range will advertise the specified cost instead of the calculated cost.

If no advertise is used in the command, the ABR prevents the selected internal routes from being summarized in a type-3 LSA and advertised to the backbone.

If no-advertise is not used in the command, the selected routes are summarized to the backbone in a type-3 LSA.

Option nssa specifies external routes (type-7 LSAs) in the configured range of route advertisements. If no-advertise (above) is used in the command, the ABR prevents the selected external routes from being summarized in a type-5 LSA and advertised to the backbone. (Configure this option where an ABR for an NSSA advertises external routes that you do not want propagated to the backbone.)

If no-advertise is not used in the command, the selected routes learned from type-7 LSAs in the area are summarized to the backbone in a type-5 LSA. [cost 1-16777215] User configured cost for an NSSA summary range. If cost is not configured, the ABR will use the algorithm defined in RFC 3101 to compute the cost and metric-type of the summarized route. If cost is specified, then the range will advertise the specified cost as the cost of the summarized route.

Assigning a cost

The cost parameter provides a way to define a fixed, user-assigned cost of an LSA type 3 summarized prefix.

Setting a summary cost to an area

This example shows how to set the summary cost to 100 for area 10 with an address range of 10.10.0.0/16.

switch(ospf)# area 10 range 10.10.0.0/16 type summary cost 100

Using a standard summary cost for an area

This example shows how to use the standard method for determining the summarized cost.
switch(ospf)# area 10 range 10.10.0.0/16 type summary

You must execute `write mem` to preserve these settings across reboots.

Setting a summary cost to an NSSA area
To set the summary cost for NSSA area 20 address range 10.20.0.0/16 to 100 with a default metric-type of type2, enter the following command.

```
switch(ospf)# area 20 range 10.20.0.0/16 type nssa cost 100
```

Setting a summary cost and metric-type to an NSSA area
To set the summary cost and metric-type for NSSA area 20 address range 10.20.0.0/16 to 100, enter the following command.

```
switch(ospf)# area 10 range 10.10.0.0/16 type nssa cost 100 metric-type type1
```

Using the RFC standard method to determine the summarized cost to an NSSA area
To change the configuration so that the 10.20.0.0/16 range uses the RFC standard method for determining the summarized cost, enter the following command.

```
switch(ospf)# area 10 range 10.10.0.0/16 type nssa
```

You must execute `write mem` to preserve these settings across reboots.

Output showing settings for summary costs
The `show ip ospf` command displays information about summary costs. An entry of `auto` indicates that the cost is calculated by the OSPF standard for summarized networks.

```
switch(config)# show ip ospf

OSPF Configuration Information
   
      Currently defined address ranges:
            Area ID   LSA Type     IP Network     Network Mask   Advertise   Cost
            --------- --------- ------------- ------------- ---------- ----
          0.0.0.10   Summary   10.10.0.0     255.255.0.0   yes         auto
          0.0.0.20   NSSA       10.20.0.0     255.255.0.0   yes         auto
          0.0.0.30   Summary   10.30.0.0     255.255.0.0   no          16777215
```

Allowing or blocking advertisement of a range of internal routes available in an area by an ABR

Defining a range of internal routes to advertise to the backbone
The commands in this example define the same range of internal routes in area 30 to summarize for injection into the backbone area. (In this example, area 30 can be a normal or stub area, or an NSSA.)

```
switch(ospf)# area 30 range 10.0.0.0/8
switch(ospf)# area 30 range 10.0.0.0/8 type summary
```

Defining a range of internal routes to block from advertising to the backbone
For the same range of routes, you can use either of the following commands to block injection of a range of summary routes (type-3 LSAs) from area 30 into the backbone.
Allowing or blocking a range of external routes available through an ASBR in an NSSA

Example of allowing or blocking a range of external RAs to the backbone

This example applies only to external routes that can be advertised from an NSSA to the backbone. The first command defines the range of external routes in the Area 7 NSSA to advertise to the backbone. The second command defines the range of external routes in the Area 7 NSSA to block from advertising to the backbone.

Influencing route choices by changing the administrative distance default

Optional: For more information, see Influencing route choices by changing the administrative distance default (optional).

Syntax:
```
distance [external | inter-area | intra-area 1-255]
```

Used in the OSPF configuration context to globally reconfigure the administrative distance priority for the specified route type.

1 is the highest priority; 255 is the lowest priority.

external 1-255
Changes the administrative distance for routes between the OSPF domain and other EGP domains.

inter-area 1-255
Changes the administrative distance for routes between areas within the same OSPF domain.

intra-area 1-255
Changes the administrative distance for routes within OSPF areas.

Default: 110; range: 1-255

Changing OSPF trap generation choices

Optional: OSPF traps (defined by RFC 1850) are supported on the routing switches. OSPF trap generation is disabled by default, but you can use the following command to enable generation of any or all of the supported OSPF traps.

Syntax:
```
trap [trap-name | all]
```

Used in the OSPF configuration context to enable or disable OSPF traps.

all
Enables or disables all OSPF traps available on the routing switch.

trap-name
Specifies a trap from table below to enable or disable.

The no form disables the specified trap.

Default: All OSPF traps disabled
The table below summarizes OSPF traps supported on the switches, and their associated MIB objects from RFC 1850.

**Table 17: Table 16: OSPF traps and associated MIB objects**

<table>
<thead>
<tr>
<th>OSPF trap name</th>
<th>MIB object</th>
</tr>
</thead>
<tbody>
<tr>
<td>interface-authentication-failure</td>
<td>ospflfAuthFailure</td>
</tr>
<tr>
<td>interface-config-error</td>
<td>ospflfConfigError</td>
</tr>
<tr>
<td>interface-receive-bad-packet</td>
<td>ospflfrxBadPacket</td>
</tr>
<tr>
<td>interface-retransmit-packet</td>
<td>ospfTxRetransmit</td>
</tr>
<tr>
<td>interface-state-change</td>
<td>ospfNbrStateChange</td>
</tr>
<tr>
<td>neighbor-state-change</td>
<td>ospfOriginateLsa</td>
</tr>
<tr>
<td>originate-lsa</td>
<td>ospfMaxAgeLsa</td>
</tr>
<tr>
<td>originate-maxage-lsa</td>
<td></td>
</tr>
<tr>
<td>virtual-interface-authentication-failure</td>
<td></td>
</tr>
<tr>
<td>virtual-interface-config-error</td>
<td>ospfVirtlfConfigError</td>
</tr>
<tr>
<td>virtual-interface-state-change</td>
<td>ospfVirtlfStateChange</td>
</tr>
<tr>
<td>virtual-neighbor-state-change</td>
<td>ospfVirtNbrStateChange</td>
</tr>
<tr>
<td>virtual-interface-receive-bad-packet</td>
<td>ospfVirtlfRxBad Packet</td>
</tr>
<tr>
<td>virtual-interface-retransmit-packet</td>
<td>ospfVirtlfTxRetransmit</td>
</tr>
</tbody>
</table>

**Example**

**Enabling OSPF traps**

If you wanted to monitor the neighbor-state-change and interface-receive-bad-packet traps, you would use the following commands to configure the routing switch to enable the desired trap. The `show` command verifies the resulting OSPF trap configuration.

```
switch(ospf)# trap neighbor-state-change
switch(ospf)# trap interface-receive-bad-packet
switch(ospf)# show ip ospf traps

OSPF Traps Enabled
====================
Neighbor State Change
Interface Receive Bad Packet
```

**Adjusting performance by changing the VLAN or subnet interface settings**

Optional: For more information, see *Adjusting performance by changing the VLAN or subnet interface settings*.

**Indicating the cost per-interface**

**Syntax:*

```
ip ospf [ip-address | all] cost 1-65535
```

Used in the VLAN context to indicate the overhead required to send a packet across an interface. You can modify the cost to differentiate between 100 Mbps and 1000 Mbps (1 Gbps) links.

Allows different costs for different subnets in the VLAN.

```
ip ospf cost 1-65535
```
Assigns the specified cost to all networks configured on the VLAN.

```
ip ospf ip-address cost 1-65535
```

Assigns the specified cost to the specified subnet on the VLAN.

```
ip ospf all cost 1-65535
```

Assigns the specified cost to all networks configured on the VLAN. (Operates the same as the `ip ospf cost` option, above.)

Default: 1; range 1-65535

**Indicating the per-interface dead interval**

**Syntax:**
```
ip ospf [ip-address | all] dead-interval 1-65535
```

Used in the VLAN context to indicate the number of seconds that a neighbor router waits for a hello packet from the specified interface before declaring the interface "down." Allows different settings for different subnet interfaces in the VLAN.

```
ip ospf dead-interval 1-65535
```

Assigns the specified dead interval to all networks configured on the VLAN.

```
ip ospf ip-address dead-interval 1-65535
```

Assigns the specified dead interval to the specified subnet on the VLAN.

```
ip ospf all dead-interval 1-65535
```

Assigns the specified dead interval to all networks configured on the VLAN. (Operates the same as the `ip ospf dead-interval option`, above.)

Default: 40 seconds; range 1-65535 seconds

**Indicating the per-interface hello interval**

**Syntax:**
```
ip ospf [ip-address | all] hello-interval 1-65535
```

Used in the VLAN context to indicate the length of time between the transmission of hello packets from the routing switch to adjacent neighbors.

The value can be from 1 to 65535 seconds. Allows different settings for different subnet interfaces in the VLAN.

```
ip ospf hello-interval 1-65535
```

Assigns the specified hello interval to all networks configured on the VLAN.

```
ip ospf ip-address hello-interval 1-65535
```

Assigns the specified hello interval to the specified subnet on the VLAN.

```
ip ospf all hello-interval 1-65535
```

Assigns the specified hello interval to all networks configured on the VLAN. Operates the same as the `ip ospf hello-interval option`.

Default: 10 seconds; range 1-65535 seconds

**Changing priority per-interface**

**Syntax:**
```
ip ospf [ip-address | all] priority 1-255
```

The priority is used when selecting the DR and backup DRs (BDRs.)

The value can be from 0 to 255 (with 255 as the highest priority.) If you set the priority to 0, the routing switch does not participate in DR and BDR election. Allows different settings for different subnet interfaces in the VLAN.

```
ip ospf priority 1-255
```
Assigns the specified priority to all networks configured on the VLAN.

```
ip ospf ip-address priority 1-255
```

Assigns the specified priority to the specified subnet on the VLAN.

```
ip ospf all priority 1-255
```

Assigns the specified priority to all networks configured on the VLAN. Operates the same as the `ip ospf priority` option.

Default: 1; range 0-255

### Changing retransmit interval per-interface

**Syntax:**

```
ip ospf [ip-address | all] retransmit-interval 0-3600
```

Used in the VLAN context to enable changing the retransmission interval for LSAs on an interface. Allows different settings for different subnet interfaces in the VLAN.

```
ip ospf priority 1-255
```

Assigns the specified retransmit interval to all networks configured on the VLAN.

```
ip ospf ip-address priority 1-255
```

Assigns the specified retransmit interval to the specified subnet on the VLAN.

```
ip ospf all priority 1-255
```

Assigns the specified retransmit interval to all networks configured on the VLAN. Operates the same as the `ip ospf priority` option.

Default: 5 seconds; range: 1-3600 seconds

### Changing transit-delay per-interface

**Syntax:**

```
ip ospf [ip-address | all] transit-delay 0-3600
```

Used in the VLAN context to enable changing the time it takes to transmit link-state update packets on this interface. Allows different settings for different subnet interfaces in the VLAN.

Default: 1 second; range: 1-3600 seconds

```
ip ospf transitch-delay 1-3600
```

Reconfigures the estimated number of seconds it takes to transmit a link-state update packet to all networks configured on the VLAN.

```
ip ospf ip-address transit-delay 1-3600
```

Reconfigures the estimated number of seconds it takes to transmit a link-state update packet to all networks configured on the specified subnet on the VLAN.

```
ip ospf all transit-delay 1-3600
```

Reconfigures the estimated number of seconds it takes to transmit a link-state update packet to all networks configured on the VLAN. (Operates the same as the `ip ospf transit-delay` option, above.)

### Examples of changing per-interface settings

Suppose that VLAN 30 is multinetted, with two subnets in area 1 and one subnet in area 5:

```
vlan 30
ip ospf 10.10.30.1 area 0.0.0.1
ip ospf 10.10.31.1 area 0.0.0.1
ip ospf 10.10.32.1 area 0.0.0.5
```
If you wanted to quickly reconfigure per-interface OSPF settings for VLAN 30, such as those listed below, you could use the commands shown in Figure

- Assign a cost of "5" to the two subnets in area 1 and a cost of "10" to the subnet in area 5.
- Assign a dead interval of 45 seconds to the subnets in area 1 and retain the default setting (40 seconds) for the subnet in area 5.

**Reconfiguring per-interface settings in a multinetted VLAN**

```markdown
<table>
<thead>
<tr>
<th>Command</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>Switch(vlan-30)# ip ospf cost 5</td>
<td>Configures a cost of &quot;5&quot; on all subnets in VLAN 30</td>
</tr>
<tr>
<td>Switch(vlan-30)# ip ospf 10.10.32.1 cost 10</td>
<td>Configures a cost of &quot;10&quot; on the 10.10.32.0 subnet in VLAN 30</td>
</tr>
<tr>
<td>Switch(vlan-30)# ip ospf dead-interval 45</td>
<td>Configures a dead interval of 45 seconds on all subnets in VLAN 30.</td>
</tr>
<tr>
<td>Switch(vlan-30)# ip ospf 10.10.32.1 dead-interval 40</td>
<td>Returns the 10.10.32.0 subnet in VLAN 30 to a default dead interval of 40</td>
</tr>
</tbody>
</table>
```

**Configuring OSPF interface authentication**

Optional: For more information, see Configuring OSPF interface authentication.

**Configuring OSPF password authentication**

**Syntax:**

```markdown
ip ospf [ip-address] authentication-key key-string
no ip ospf [ip-address] authentication
```

Used in the VLAN interface context to configure password authentication for all interfaces in the VLAN or for a specific subnet. The password takes effect immediately, and all OSPF packets transmitted on the interface contain this password. All OSPF packets received on the interface are also checked for the password. If it is not present, the packet is dropped.

To disable password authentication on an interface, use the no form of the command.

When the switch is in enhanced secure mode, commands that take a secret key as a parameter have the echo of the secret typing replaced with asterisks. The input for key-string is prompted for interactively. For more information, see the access security guide for your switch.

- **ip-address**
  
  Used in subnetted VLAN contexts where you want to assign or remove a password associated with a specific subnet.

  Omit this option when you want the command to apply to all interfaces configured in the VLAN.

- **key-string**
  
  An alphanumeric string of one to eight characters. (Spaces are not allowed.)

  To change the password, re-execute the command with the new password.

  Use `show ip ospf interface ip-address` to view the current authentication setting.

**Note:** To replace the password method with the MD5 method on a given interface, overwrite the password configuration by using the MD5 form of the command shown in the next syntax description. (It is not necessary to disable the currently configured OSPF password.)

**Default:** Disabled
Configuring OSPF MD5 authentication

Syntax:

```
ip ospf md5-auth-key-chain chainname-string
no ip ospf [ip-address] authentication
```

Used in the VLAN interface context to configure MD5 authentication for all interfaces in the VLAN or for a specific subnet. The MD5 authentication takes effect immediately, and all OSPF packets transmitted on the interface contain the designated key. All OSPF packets received on the interface are also checked for the key. If it is not present, the packet is dropped.

To disable MD5 authentication on an interface, use the `no` form of the command.

---

**NOTE**

Before using this authentication option, you must configure one or more key chains on the routing switch by using the Key Management System (KMS). See the access security guide for your switch.

Default: Disabled

**ip-address**

Used in subnetted VLAN contexts where you want to assign or remove MD5 authentication associated with a specific subnet.

Omit this option when you want the command to apply to all interfaces configured in the VLAN.

**chain-name-string**

The name of a key generated using the `key-chain chain_name key key_id`.

To change the MD5 authentication configured on an interface, re-execute the command with the new MD5 key.

Use `show ip ospf interface ip-address` to view the current authentication setting.

---

**NOTE**

To replace the MD5 method with the password method on a given interface, overwrite the MD5 configuration by using the password form of the command shown in the next syntax description. (It is not necessary to disable the currently configured OSPF MD5 authentication.)

Default: Disabled

---

**Configuring a virtual link**

For information about virtual links, see Configuring an ABR to use a virtual link to the backbone.

Syntax:

```
ip ospf area area-id virtual link ip-address
```

Used on a pair of ABRs at opposite ends of a virtual link in the same area to configure the virtual link connection.

**area-id**

This must be the same for both ABRs in the link and is the area number of the virtual link transit area in either decimal or dotted decimal format.

**ip-address**

On an ABR directly connected to the backbone area, this value must be the IP address of an ABR (in the same area) needing a virtual link to the backbone area as a substitute for a direct physical connection.

On the ABR that needs the virtual link to the backbone area, this value must be the IP address of the ABR (in the same area) having a direct physical connection to the backbone area.

**Example**
shows an OSPF ABR, routing switch "A" that lacks a direct connection to the backbone area (area 0.) To provide backbone access to routing switch "A," you can add a virtual link between routing switch "A" and routing switch "C," using area 1 as a transit area.

To configure the virtual link, define it on the routers that are at each end of the link. No configuration for the virtual link is required on the other routers on the path through the transit area (such as routing switch "B" in this example.)

**Defining OSPF virtual links within a network**

To configure the virtual link on routing switch "A," enter the following command specifying the area 1 interface on routing switch "C":

```
switch(ospf)# area 1 virtual-link 209.157.22.1
```

To configure the virtual link on routing switch "C," enter the following command specifying the area 1 interface on routing switch "A":

```
switch(ospf)# area 1 virtual-link 10.0.0.1
```

For descriptions of virtual link interface parameters you can either use in their default settings or reconfigure as needed, see [Changing the dead interval on a virtual link](#).

**Changing the dead interval on a virtual link**

For more information, see [Adjusting virtual link performance by changing the interface settings](#).

**Syntax:**

```
area area-id virtual link ip-address dead-interval 1-65535
```

Used in the router OSPF context on both ABRs in a virtual link to change the number of seconds that a neighbor router waits for a hello packet from the specified interface before declaring the interface "down." This should be some multiple of the hello interval. The `dead-interval` setting must be the same on both ABRs on a given virtual link.

- **area-id**
  - Specifies the OSPF area in which both ABRs in a given virtual link operate. In this use, the area ID is sometimes termed "transit area ID."
  - This value must be the same for both ABRs in the virtual link.
- **ip-address**
  - For an ABR in a given virtual link, this is the IP address used to create the link on that ABR.
This IP address matches the IP address of the interface on the opposite end of the virtual link. See the description of ip-address in the syntax description under Configuring a virtual link.
Use show ip ospf virtual-link  ip-address to view the current setting.
Default: 40 seconds; range: 1-65535 seconds

**Indicating the hello interval on a virtual link**

**Syntax:**
area area-id virtual link ip-address hello-interval 1-65535

Used in the router OSPF context on both ABRs in a virtual link to indicate the length of time between the transmission of hello packets between the ABRs on opposite ends of the virtual link.
The hello-interval setting must be the same on both ABRs on a given virtual link.
Default: 10 seconds; range: 1-65535 seconds
area-id

Specifies the OSPF area in which both ABRs in a given virtual link operate. In this use, the area ID is sometimes termed "transit area ID."
This value must be the same for both ABRs in the virtual link.
/ip-address

For an ABR in a given virtual link, this is the IP address used to create the link on that ABR. (This IP address matches the IP address of the interface on the opposite end of the virtual link. See the description of ip-address in the syntax description under Configuring a virtual link.)
Use show ip ospf virtual-link  ip-address to view the current setting.

**Changing the retransmitting interval on a virtual link**

**Syntax:**
area area-id virtual link ip-address retransmit-interval 1-3600

Used in the router OSPF context on both ABRs in a virtual link to change the number of seconds between LSA retransmissions on the virtual link.
The retransmit-interval setting must be the same on both ABRs on a given virtual link. This value is also used when retransmitting database description and link-state request packets.
area-id

Specifies the OSPF area in which both ABRs in a given virtual link operate. In this use, the area ID is sometimes termed "transit area ID." This value must be the same for both ABRs in the virtual link.
/ip-address

For an ABR in a given virtual link, this is the IP address used to create the link on that ABR. (This IP address matches the IP address of the interface on the opposite end of the virtual link. See the description of ip-address in the syntax description under Configuring a virtual link.)
Use show ip ospf virtual-link  ip-address to view the current setting.
Default: 5 seconds; range: 1-3600 seconds

**Changing the transit-delay on a virtual link**

**Syntax:**
area area-id virtual link ip-address transit-delay [0-3600]

Used in the router OSPF context on both ABRs in a virtual link to change the estimated number of seconds it takes to transmit a link state update packet over a virtual link. The transit-delay setting must be the same on both ABRs on a given virtual link.
area-id
Specifies the OSPF area in which both ABRs in a given virtual link operate. In this use, the area ID is sometimes termed "transit area ID." This value must be the same for both ABRs in the virtual link.

ip-address
For an ABR in a given virtual link, this is the IP address used to create the link on that ABR. (This IP address matches the IP address of the interface on the opposite end of the virtual link. See the description of ip-address in the syntax description under Configuring a virtual link.)

Use show ip ospf virtual-link ip-address to view the current setting.

Default: 1 second; range: 1-3600 seconds

Example
To change the hello-interval on the virtual link configured for the network in Configuring a virtual link to 60 seconds:

- On routing switch "A" (IP address 10.0.0.1) you would use the following command to reconfigure the current hello-interval to 60 seconds:

  ```
  switch(ospf)# area 1 virtual-link 209.157.22.1 hello-interval 60
  ```

- On routing switch "C" (IP address 209.157.22.1) you would use the following command to reconfigure the current hello-interval to 60 seconds:

  ```
  switch(ospf)# area 1 virtual-link 10.0.0.1 hello-interval 60
  ```

Configuring OSPF authentication on a virtual link
For more information, see Configuring OSPF authentication on a virtual link.

Authenticating OSPF MD5 on a virtual link

Syntax:

```
ip ospf md5-auth-key-chain chain-name-string no ip ospf [ip-address] authentication
```

Used to configure MD5 authentication in the router OSPF context on both ABRs in a virtual link. The MD5 authentication takes effect immediately, and all OSPF packets transmitted on the link contain the designated key. Every OSPF packet received on the interface for the virtual link on each ABR is checked for the key. If it is not present, the packet is dropped.

To disable MD5 authentication on an ABR interface used for a virtual link, use the no form of the command. The password must be the same on both ABRs on a given virtual link.
Before using this authentication option, you must configure one or more key chains on the routing switch by using the Key Management System (KMS). See the Access Security Guide for your routing switch.

**ip-address**

For an ABR in a given virtual link, this is the IP address used to create the link on that ABR. (This IP address matches the IP address of the interface on the opposite end of the virtual link. See the description of **ip-address** in the syntax description under Configuring a virtual link.)

**chain-name-string**

The name of a key generated using the key-chain chain_name key key_id command.

To change the MD5 authentication configured on a virtual link, re-execute the command with the new MD5 key.

To replace the MD5 method with the password method on a virtual link, overwrite the MD5 configuration by using the password form of the command. (It is not necessary to disable the currently configured OSPF MD5 authentication.)

Default: Disabled

---

## Configuring a passive OSPF interface

For more information, see About OSPF passive.

Enter this command in VLAN context:

```
switch(vlan-1)# ip ospf passive
```

**Syntax:**

- `ip ospf ip-addr passive`
- `no ip ospf ip-addr passive`

Configures passive OSPF for an AS.

`ip-addr`

Optionally, you can configure an IP address on the VLAN.

The `no` option disables the passive option; the interface becomes an active interface.

Default: Active

**Example**

To display the OSPF passive information, enter the command as shown:

```
show ip ospf interface command with passive configured on an interface
```

```
switch(vlan-1)# show ip ospf interface

<table>
<thead>
<tr>
<th>IP Address</th>
<th>Status</th>
<th>Area ID</th>
<th>State</th>
<th>Auth-type</th>
<th>Cost</th>
<th>Priority</th>
<th>Passive</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.10.10.1</td>
<td>enabled</td>
<td>0.0.0.2</td>
<td>down</td>
<td>none</td>
<td>1</td>
<td>1</td>
<td>Yes</td>
</tr>
<tr>
<td>10.12.13.1</td>
<td>enabled</td>
<td>0.0.0.2</td>
<td>wait</td>
<td>none</td>
<td>1</td>
<td>1</td>
<td>No</td>
</tr>
</tbody>
</table>
```
show ip ospf interface command for a specific VLAN with passive configured on an interface

```
switch(config) show ip ospf interface vlan 4

OSPF configuration and statistics for VLAN 4

OSPF Interface Status for 10.10.10.1

IP Address: 10.10.10.1 Status: enabled
AreaID: 0.0.0.2 Passive: Yes

State: DOWN Auth-type: none
Cost: 1 Chain: 
Type: BCAST Priority: 1

Transit Delay: 1 Retrans Interval: 5
Hello Interval: 10 Rtr Dead Interval: 40
Designated Router: Events: 0
Backup Desig. Rtr: Passive: yes
Neighbors: 2

Configuring OSPF Network Type

Syntax:
ip ospf network-type
no ip ospf network-type

Allows you to configure an interface with one of the following OSPF network types:

- **Point-to-point**: Sets the network type of an interface to point-to-point.
- **BCAST**: Sets the network type of an interface to broadcast.

The `no` option disables the OSPF network type for an interface.

Example

To display the OSPF network-type information, enter the command as shown in the following example:

```
sh ip ospf command with network-type configured on an interface

switch(config) sh ip ospf network-type

OSPF Interface Status for 2.1.1.1

IP Address: 2.1.1.1 Status: enabled
AreaID: backbone Passive: Yes

State: DOWN Auth-type: none
Cost: 1 Chain: 
Type: Point-to-point Priority: n/a

Transit Delay: 1 Retrans Interval: 5
Hello Interval: 10 Rtr Dead Interval: 40
Designated Router: Events: 0
Backup Desig. Rtr: Passive: no
Neighbors: 0
BFD Administrative Status: Disabled
```

Configuring the calculation interval
Syntax:
```
```
Enables and configures SPF scheduling (throttling.) This delays SPF calculations during periods of network topology changes.

SPF calculations occur at the interval set by the `spf-throttle` command. This command is executed in `ospf` context.

Default: 5 seconds
```
start-interval [1-600]
```
Specifies the initial SPF schedule delay in seconds.
```
wait-interval [1-600]
```
specifies the amount of time to wait until the next SPF calculation occurs, in seconds.
```
max-wait-time [1-600]
```
Specifies the maximum time between two consecutive SPF calculations, in seconds.

The current SPF interval is calculated; it will be twice as long as the previous interval until this value reaches the maximum-wait-time specified.

Example

**SPF throttling configuration**

The last SPF calculation was scheduled and triggered at the 100th second. A new topology event occurred at the 104th second. The configured values are:

- start-interval
  - = 3 seconds
- wait-interval
  - = 3 seconds
- max-wait-time
  - = 500 seconds

```
switch(ospf)# spf-throttle start-interval 3 wait-interval 3 max-wait-time 500
```

- The difference between the last SPF (100), added to the current SPF throttle interval (3), is less than the time of the occurrence of the network event (104.) SPF is scheduled to run instantly and the current SPF throttle interval is configured to 3 seconds (the start-interval value.)
- Another topology event occurs within the above 3 second SPF throttle interval, at the 106th second. SPF is scheduled to run again at the 107th second (last event at 104th second+ wait-interval of 3 seconds), which is greater than the current event (106th second.) The SPF timer is scheduled to run after 1 second. After that, the current SPF throttle interval is changed to 10 seconds, the current wait-interval value.

If another topology event occurs at the 110th second, which is within the 10 second current wait-time. SPF is scheduled to run again at the 117th second (last SPF of 107 seconds + wait-interval of 10 seconds), which is greater than the current event (110 seconds.) The SPF timer is scheduled to run after 7 seconds. The current SPF wait-time is doubled to 20 seconds.

If any topology event occurs during the dynamic wait-interval, SPF is scheduled according to the formula:

Last SPF + current dynamic wait-interval - time of occurrence of the event
The dynamic wait-interval keeps doubling until the max-wait-time is reached. If the max-wait-time is reached and the network continues to be unstable, the dynamic wait-time is set to the max-wait-time until the network stabilizes.

If the network stabilizes during a dynamic wait-interval period, SPF is calculated immediately and the current SPF wait-interval is set to the configured start-interval.

### Viewing OSPF information

### Viewing general OSPF configuration information

**Syntax:**

```
show ip ospf general
```

**General output for the show ip ospf command**

```
switch(config)# show ip ospf general

OSPF General Status
OSPF protocol : enabled
Router ID : 17.255.134.231
RFC 1583 compatability : compatible
Intra-area distance : 110
Inter-area distance : 110
AS-external distance : 110
Default import metric : 10
Default import metric type : external type 2
Area Border : no
AS Border : no
External LSA Count : 0
External LSA Checksum Sum : 0
Originate New LSA Count : 0
Receive New LSA Count : 0
Graceful Restart Interval : 120
Graceful Restart Strict-Lsa Checking : Enabled
Nonstop forwarding : Disabled
Log Neighbor Adjacency Changes : Enabled

SPF Throttling
Start Interval : 3
Wait Interval : 3
Maximum Wait Time : 500
Current Wait Interval : 3
```

The `show running-config` command also displays the SPF configuration information. The configured parameters for SPF are highlighted in bold below.

```
switch(config)# show running-config

Running configuration:

; J8693A Configuration Editor; Created on release #K.15.07.0000x
```
hostname "switch"
module 1 type J86yyA
module 2 type J86xxA
vlan 1
  name "DEFAULT_VLAN"
  untagged 1-4,7-48,A1-A4
  ipv6 address fe80::2 link-local
  ip address dhcp-bootp
  ipv6 enable
  no untagged 5-6
  exit
power-over-ethernet pre-std-detect
router ospf
  spf-throttle start-interval 3 wait-interval 3 max-wait-time 500
  exit
snmp-server community "public" unrestricted

The following fields are shown in the OSPF general status display:

<table>
<thead>
<tr>
<th>Field</th>
<th>Content</th>
</tr>
</thead>
<tbody>
<tr>
<td>OSPF protocol</td>
<td>Whether OSPF is currently enabled.</td>
</tr>
<tr>
<td>Router ID</td>
<td>Router ID that this routing switch is currently using to identify itself.</td>
</tr>
<tr>
<td>RFC 1583 compatibility</td>
<td>Whether the routing switch is currently using RFC 1583 (compatible) or RFC 2328 (non-compatible rules for calculating external routes).</td>
</tr>
<tr>
<td>Intra-area distance</td>
<td>Administrative distance for routes within OSPF areas.</td>
</tr>
<tr>
<td>Inter-area distance</td>
<td>Administrative distance for routes between areas within the same OSPF domain.</td>
</tr>
<tr>
<td>AS-external</td>
<td>Administrative distance for routes between the OSPF domain and other, Exterior Gateway Protocol domains.</td>
</tr>
<tr>
<td>Default import metric</td>
<td>Default metric that will be used for any routes redistributed into OSPF by this routing switch</td>
</tr>
<tr>
<td>Default import metric type</td>
<td>Metric type (type 1 or type 2) that will be used for any routes redistributed into OSPF by this routing switch</td>
</tr>
<tr>
<td>Area Border</td>
<td>Whether this routing switch is currently acting as an area border router.</td>
</tr>
<tr>
<td>AS Border</td>
<td>Whether this routing switch is currently acting as an AS border router (redistributing routes.)</td>
</tr>
<tr>
<td>External LSA Count</td>
<td>Total number of external LSAs currently in the routing switch's link state database.</td>
</tr>
</tbody>
</table>
### Viewing OSPF area information

**Syntax:**

```
show ip ospf area [ospf-area-id]
```

The `[ospf-area-id]` parameter shows information for the specified area. If no area is specified, information for all the OSPF areas configured is displayed.

The OSPF area display shows the information found in the table:

**Table 19: CLI display of OSPF area information**

<table>
<thead>
<tr>
<th>Field</th>
<th>Content</th>
</tr>
</thead>
<tbody>
<tr>
<td>Area ID</td>
<td>Identifier for this area.</td>
</tr>
<tr>
<td>Type</td>
<td>Area type, which can be either &quot;normal&quot; or &quot;stub&quot;.</td>
</tr>
<tr>
<td>Cost</td>
<td>Metric for the default route that the routing switch will inject into a stub area if the routing switch is an ABR for the area. This value applies only to stub areas.</td>
</tr>
<tr>
<td>SPFR</td>
<td>Number of times the routing switch has run the shortest path first route calculation for this area.</td>
</tr>
<tr>
<td>ABR</td>
<td>Number of area border routers in this area.</td>
</tr>
<tr>
<td>ASBR</td>
<td>Number of autonomous system border routers in this area.</td>
</tr>
<tr>
<td>LSA</td>
<td>Number of LSAs in the link state database for this area.</td>
</tr>
</tbody>
</table>
Field | Content
--- | ---
Chksum(Hex) | Sum of the checksums of all LSAs currently in the area's link state database. This value can be compared to the value for other routers in the area to verify database synchronization.

**Example**

**show ip ospf area output**

```plaintext
switch(config)# show ip ospf area

OSPF Area Information

<table>
<thead>
<tr>
<th>Area ID</th>
<th>Type</th>
<th>Cost</th>
<th>SPFR</th>
<th>ABR</th>
<th>ASBR</th>
<th>LSA</th>
<th>Checksum</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.0.0.0</td>
<td>normal</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0x0000781f</td>
</tr>
<tr>
<td>192.147.60.0</td>
<td>normal</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0x0000fee6</td>
</tr>
<tr>
<td>192.147.80.0</td>
<td>stub</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>0x000181cd</td>
</tr>
</tbody>
</table>
```

**Viewing OSPF external link-state information**

**Syntax:**

`show ip ospf external-link-state`

When you enter this command, an output similar to the following is displayed:

**Output for the show ip ospf external-link-state command**

```plaintext
switch# show ip ospf external-link-state

OSPF External LSAs

<table>
<thead>
<tr>
<th>Link State ID</th>
<th>Router ID</th>
<th>Age</th>
<th>Sequence #</th>
<th>Checksum</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.3.7.0</td>
<td>10.0.8.37</td>
<td>232</td>
<td>0x80000005</td>
<td>0x0000d99f</td>
</tr>
<tr>
<td>10.3.8.0</td>
<td>10.0.8.37</td>
<td>232</td>
<td>0x80000005</td>
<td>0x0000ce9a</td>
</tr>
<tr>
<td>10.3.9.0</td>
<td>10.0.8.37</td>
<td>232</td>
<td>0x80000005</td>
<td>0x0000c3b3</td>
</tr>
<tr>
<td>10.3.10.0</td>
<td>10.0.8.37</td>
<td>232</td>
<td>0x80000005</td>
<td>0x0000b8bd</td>
</tr>
<tr>
<td>10.3.33.0</td>
<td>10.0.8.36</td>
<td>1098</td>
<td>0x800009cd</td>
<td>0x0000b9dd</td>
</tr>
</tbody>
</table>
```

The following table shows the information the OSPF external link state displays:

**Table 20: CLI display of OSPF external link state information**

<table>
<thead>
<tr>
<th>Field</th>
<th>Content</th>
</tr>
</thead>
<tbody>
<tr>
<td>Link State ID</td>
<td>LSA ID for this LSA. Normally, the destination of the external route, but may have some &quot;host&quot; bits set.</td>
</tr>
<tr>
<td>Router ID</td>
<td>Router ID of the router that originated this external LSA.</td>
</tr>
<tr>
<td>Age</td>
<td>Current age (in seconds) of this LSA.</td>
</tr>
</tbody>
</table>
### Syntax:

**show ip ospf external-link-state [status] [subset-options]**
- `router-id ip-addr`  
  Subset option to filter displayed external-link-state data to show LSAs with the specified router ID only. Can also be filtered by using the `link-state-id` or `sequence-number` options.
- `sequence-number integer`  
  Subset option to filter displayed external-link-state data to show LSAs with the specified sequence number. Can also be filtered by using the `link-state-id` or `router-id` options.
- `link-state-id ip-addr`  
  Subset option to filter displayed external-link-state data to show LSAs with the specified ID only. Can also be filtered by using the `sequence-number` or `router-id` options.

**Syntax:**

**show ip ospf external-link-state [status] advertise**
Displays the hexadecimal data in the specified LSA packet, the actual contents of the LSAs. Can also be filtered by using the `link-state-id`, `router-id`, or `sequence-number` options.

### Output for show ip ospf external-link-state advertise

```
switch# show ip ospf external-link-state advertise
OSPF External LSAs
Advertisements
-----------------------------------------------
000302050a0307000a0008258000000005d99f0024ffffff008000000a0000000000000000
000302050a0308000a0008258000000005ce9a9024ffffff008000000a0000000000000000
000302050a0309000a0008258000000005c3b30024ffffff008000000a0000000000000000
000302050a030a000a0008258000000005b8bd0024ffffff008000000a0000000000000000
000302050a0321000a000824800009cdb9dd024ffffff00800000010000000000000000
```

### Viewing OSPF interface information

**Syntax:**

**show ip ospf interface [vlan vlan-id | ip-addr]**
- `ip-address`  
  Displays the OSPF interface information for the specified IP address.
- `vlan-id`  
  Displays the OSPF interface information for the specified IP address.

The following table shows the information displayed for the OSPF interface.

<table>
<thead>
<tr>
<th>Field</th>
<th>Content</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sequence #</td>
<td>Sequence number of the current instance of this LSA.</td>
</tr>
<tr>
<td>Chksum(Hex)</td>
<td>LSA checksum value.</td>
</tr>
</tbody>
</table>
**Table 21: CLI display of OSPF interface information**

<table>
<thead>
<tr>
<th>Field</th>
<th>Content</th>
</tr>
</thead>
<tbody>
<tr>
<td>IP Address</td>
<td>The local IP address for this interface.</td>
</tr>
<tr>
<td>Status</td>
<td>enabled or disabled—Whether OSPF is currently enabled on this interface.</td>
</tr>
<tr>
<td>Area ID</td>
<td>The ID of the area that this interface is in.</td>
</tr>
<tr>
<td>State</td>
<td>The current state of the interface. The value will be one of the following: <strong>DOWN</strong>&lt;br&gt;The underlying VLAN is down.&lt;br&gt;<strong>WAIT RCC</strong>&lt;br&gt;The underlying VLAN is up, but we are waiting to hear hellos from other routers on this interface before we run designated router election.&lt;br&gt;<strong>Pt-to-Pt</strong>&lt;br&gt;When network interface is point-to-point, DR, BDR and Priority fields are 'n/a' and State and Type should be 'point-to-point'.&lt;br&gt;<strong>.DR</strong>&lt;br&gt;This switch is the designated router for this interface.&lt;br&gt;<strong>BDR</strong>&lt;br&gt;This switch is the backup designated router for this interface.&lt;br&gt;<strong>DROther</strong>&lt;br&gt;This router is not the designated router or backup designated router for this interface.</td>
</tr>
<tr>
<td>Auth-type</td>
<td>none&lt;br&gt;or&lt;br&gt;simple&lt;br&gt;— Will be none&lt;br&gt;if no authentication key is configured,&lt;br&gt;simple&lt;br&gt;if an authentication key is configured. All routers running OSPF on the same link must be using the same authentication type and key.</td>
</tr>
<tr>
<td>Chain</td>
<td>The name of the key chain configured for the specified interface. (See the access security guide for your switch.)</td>
</tr>
<tr>
<td>Cost</td>
<td>The OSPF's metric for this interface.</td>
</tr>
<tr>
<td>Pri</td>
<td>This routing switch's priority on this interface for use in the designated router election algorithm.</td>
</tr>
<tr>
<td>Passive</td>
<td>Whether the interface sends link-state advertisements (LSAs) to all other routers in the same Autonomous System (AS.)</td>
</tr>
</tbody>
</table>
Example

Output for show ip ospf interface

```
switch# show ip ospf interface
OSPF Interface Status

<table>
<thead>
<tr>
<th>IP Address</th>
<th>Status</th>
<th>Area ID</th>
<th>State</th>
<th>Auth-type</th>
<th>Cost</th>
<th>Pri</th>
<th>Passive</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.3.18.36</td>
<td>enabled</td>
<td>10.3.16.0</td>
<td>DOWN</td>
<td>none</td>
<td>1</td>
<td>1</td>
<td>no</td>
</tr>
<tr>
<td>10.3.53.36</td>
<td>enabled</td>
<td>10.3.48.0</td>
<td>BDR</td>
<td>none</td>
<td>1</td>
<td>1</td>
<td>no</td>
</tr>
</tbody>
</table>
```

OSPF interface configuration

```
<table>
<thead>
<tr>
<th>IP Address</th>
<th>Area ID</th>
<th>Admin Status</th>
<th>Type</th>
<th>Authen Type</th>
<th>Cost</th>
<th>Pri</th>
<th>----</th>
<th>---</th>
</tr>
</thead>
<tbody>
<tr>
<td>192.16.30.186</td>
<td>backbone</td>
<td>enabled</td>
<td>Pt-to-Pt</td>
<td>none</td>
<td>100</td>
<td>n/a</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
```

Viewing OSPF interface information for a specific VLAN or IP address

**Syntax:**
```
show ip ospf interface [vlan vlan-id | ip-addr]
```

To display OSPF interface information for a specific VLAN or IP address, enter the `show ip ospf interface ip-addr` command at any CLI level.

**Table 22: CLI display of OSPF interface information—VLAN or IP address**

<table>
<thead>
<tr>
<th>Field</th>
<th>Content</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type</td>
<td>BCAST or Point-to-point. NBMA (frame relay or ATM) type interfaces are not supported on the switches.</td>
</tr>
<tr>
<td>Transit Delay</td>
<td>Configured transit delay for this interface.</td>
</tr>
<tr>
<td>Retrans Interval</td>
<td>Configured retransmit interval for this interface.</td>
</tr>
<tr>
<td>Hello Interval</td>
<td>Configured hello interval for this interface.</td>
</tr>
<tr>
<td>Rtr Dead Interval</td>
<td>Configured router dead interval for this interface.</td>
</tr>
<tr>
<td>Network Type</td>
<td>Configured network type for this interface, BCAST or Point-to-point.</td>
</tr>
<tr>
<td>Designated Router</td>
<td>IP address of the router that has been elected DR on this interface.</td>
</tr>
<tr>
<td>Backup Desig. Rtr</td>
<td>IP address of the router that has been elected BDR on this interface.</td>
</tr>
<tr>
<td>Events</td>
<td>Number of times the interface state has changed.</td>
</tr>
</tbody>
</table>
### Field | Content
--- | ---
Passive | Whether the interface sends LSAs to all other routers in the same Autonomous System (AS.)
Neighbors | Number of neighbors.

If you use `show ip ospf interface vlan vlan-id`, the output is the same as shown in the previous table, except for the IP address on the indicated VLAN.

#### Examples

**show ip ospf interface ip-addr output**

```
switch(ospf)# sho ip ospf int 10.10.50.1

OSPF Interface Status for 10.3.1836
IP Address : 10.3.18.36 Status : enabled
Area ID : 10.3.16.0
State : BDR Auth-type : none
Cost : 1 Chain :
Type : BCAST Priority : 1
Transit Delay : 1 Retrans Interval : 5
Hello Interval : 10 Rtr Dead Interval : 40
Designated Router : 10.3.18.34 Events : 3
Backup Desig. Rtr : 10.3.18.36
Backup Desig. Rtr : 10.3.18.36
```

**show ip ospf interface ip-addr backbone output**

```
switch# show ip ospf interface 10.2.1.2

OSPF Interface Status for 10.2.1.2
IP Address : 10.2.1.2 Status : enabled
Area ID : backbone
State : Point-to-point Auth-type : none
Cost : 1 Chain :
Type : Point-to-point Priority : n/a
Transit Delay : 1 Retrans Interval : 5
Hello Interval : 10 Rtr Dead Interval : 40
Designated Router : n/a Events : 0
Backup Desig. Rtr : n/a Passive : no
Neighbors : 1
```

**show ip ospf interface ip-addr neighbor output**

```
switch# show ip ospf neighbor

OSPF Neighbor Information

<table>
<thead>
<tr>
<th>Router</th>
<th>Pri</th>
<th>IP Address</th>
<th>NbIfState</th>
<th>State</th>
<th>Rxmt QLen</th>
<th>Events</th>
<th>Helper Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1.1.1</td>
<td>n/a</td>
<td>10.2.1.1</td>
<td>n/a</td>
<td>FULL</td>
<td>0</td>
<td>6</td>
<td>None</td>
</tr>
</tbody>
</table>
```
**show ip ospf neighbor detail**

```
switch# show ip ospf neighbor detail

OSPF Neighbor Information for neighbor 10.2.1.2

    IP Address : 10.2.1.2
    Router ID  : 2.2.2.2
    Interface  : vlan-1
    Area       : backbone
    Priority   : n/a
    Options    : 0x42
    Events     : 7

    State          : FULL
    Designated Router : n/a
    Backup Designated Router : n/a
    Retransmit Queue Length : 0
    Neighbor Uptime : 0h:0m:14s
    Dead Timer Expires : 35 sec
```

**Show ip ospf neighbor detail**

```
switch# show ip ospf neighbor 10.2.1.2

OSPF Neighbor Information for neighbor 10.2.1.2

    IP Address : 10.2.1.2
    Router ID  : 2.2.2.2
    NbIfState  : n/a
    Rxtm QLen  : 0
    Helper Status : None

    Pri : n/a
    State : FULL
    Events : 7
    Helper Age : 0
```

**show ospf interface configuration**

```
<table>
<thead>
<tr>
<th>IP Address</th>
<th>Area ID</th>
<th>Status</th>
<th>Type</th>
<th>Type</th>
<th>Cost</th>
<th>Pri</th>
</tr>
</thead>
<tbody>
<tr>
<td>172.16.30.186</td>
<td>backbone</td>
<td>enabled</td>
<td>Pt-to-Pt</td>
<td>none</td>
<td>100</td>
<td>n/a</td>
</tr>
</tbody>
</table>
```

**Viewing OSPF packet statistics for a subnet or VLAN**

Displays the statistics on OSPF packets sent and received on the interfaces in VLANs and/or subnets on an OSPF-enabled routing switch, including the number of errors that occurred during packet transmission. Enter the command at any CLI level.

**Syntax:**

```
show ip ospf interface [[vlan vlan-id] | ip-address]
```

Displays the following information for OSPF-enabled VLANs and/or subnets:

- **vlan-id**
  - Displays OSPF packet statistics for all subnets configured on the VLAN.
- **ip-address**
  - Displays OSPF packet statistics only for a specified VLAN subnet.

**Displaying OSPF statistics for VLAN traffic**

```
switch(ospf)# show ip ospf statistics vlan 1

OSPF statistics for VLAN 1

OSPF Interface Status for 10.0.0.2
```
Tx Hello Packet Count : 16  Rx Hello Packet Count : 16
Tx DD Packet Count : 2Rx DD Packet Count : 4
Tx LSR Packet Count : 1  Rx LSR Packet Count : 1
Tx LSU Packet Count : 5  Rx LSU Packet Count : 2
Tx LSA Packet Count : 2  Rx LSA Packet Count : 3

OSPF Errors: 26

**Table 23: CLI display of OSPF statistics for VLAN traffic**

<table>
<thead>
<tr>
<th>Field</th>
<th>Content</th>
</tr>
</thead>
<tbody>
<tr>
<td>OSPF statistics for VLAN vlan-id</td>
<td>OSPF statistics displayed for the specified VLAN number.</td>
</tr>
<tr>
<td>OSPF Interface Status for ip-address</td>
<td>IP address of a subnet on the VLAN.</td>
</tr>
<tr>
<td>Tx/Rx Hello Packet Count</td>
<td>Number of OSPF hello packets sent/received on each subnet interface.</td>
</tr>
<tr>
<td>Tx/Rx DD Packet Count</td>
<td>Number of link-state database description packets sent/received on each subnet interface.</td>
</tr>
<tr>
<td>Tx/Rx LSR Packet Count</td>
<td>Number of link-state request packets sent/received on each subnet interface.</td>
</tr>
<tr>
<td>Tx/Rx LSU Packet Count</td>
<td>Number of link-state update packets sent/received on each subnet interface.</td>
</tr>
<tr>
<td>Tx/Rx LSA Packet Count</td>
<td>Number of link-state acknowledgement packets sent/received on each subnet interface.</td>
</tr>
<tr>
<td>OSPF errors</td>
<td>Number of errors detected on the VLAN subnet during OSPF packet exchange.</td>
</tr>
</tbody>
</table>

**Displaying OSPF statistics for subnet traffic**

```
switch(ospf)# show ip ospf statistics 10.0.0.2
```

**OSPF Interface Statistics**

<table>
<thead>
<tr>
<th>IP Address</th>
<th>Total Tx</th>
<th>Total Rx</th>
<th>Total Errors</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0.0.2</td>
<td>15</td>
<td>15</td>
<td>15</td>
</tr>
</tbody>
</table>

**Table 24: CLI display of OSPF statistics for VLAN subnet traffic**

<table>
<thead>
<tr>
<th>Field</th>
<th>Content</th>
</tr>
</thead>
<tbody>
<tr>
<td>IP Address</td>
<td>IP address of subnet.</td>
</tr>
</tbody>
</table>
### Per-subnet OSPF statistics

<table>
<thead>
<tr>
<th>Field</th>
<th>Content</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total Tx</td>
<td>Total number of OSPF packets sent on each subnet interface.</td>
</tr>
<tr>
<td>Total Rx</td>
<td>Total number of OSPF packets received on each subnet interface.</td>
</tr>
<tr>
<td>Total Errors</td>
<td>Total number of errors in OSPF packet transmission on each subnet interface.</td>
</tr>
</tbody>
</table>

### Clearing OSPF statistics for all VLAN interfaces on the switch

**Syntax:**

clear ip ospf statistics

Clears the OSPF statistics for all VLAN interfaces on the switch and sets all VLAN/subnet counters for OSPF traffic to zero. Enter the command at any CLI level.

### Viewing OSPF link-state information

**Syntax:**

show ip ospf link-state [status] [subsetoptions] [advertise [subset-options]] [detail]

To display OSPF link state information, enter `show ip ospf link-state` at any CLI level.

#### advertise

Displays the hexadecimal data in LSA packets (advertisements) for the OSPF areas configured on the routing switch.

The output can also be filtered by area (area-id), link-state-id, router-id, sequence-number, and/or type.

Default: All OSPF areas configured on the routing switch.

#### ospf-area-id

Used to restrict display of LSA database or advertisements to show only the data from a specific OSPF area.

Can also be used with other subset options (router-id, sequence-number, external link-state-id, and/or type) to further define the source of displayed information.

#### link-state-id ip-addr

Used to restrict display of LSA database or advertisements to show only the data from sources having the specified IP address as a link-state ID.

Can also be used with other subset options (ospf-area-id, router-id, sequence-number, external link-state-id, and type) to further define the source of displayed information.

#### router-id ip-addr

Used to restrict display of LSA database or advertisements to show only the data from sources having the specified router ID.

Can also be used with other subset options (ospf-area-id, link-state-id, sequence-number, and type) to further define the source of displayed information.

#### sequence-number integer

Used to restrict display of LSA database or advertisements to show only the data from sources having the specified sequence number.
Can also be used with other subset options ( ospf-area-id, link-state-id, router-id, and type) to further define the source of displayed information.

**type [router | network | summary | as-summary | external | multicast | nssa]**

Used to restrict display of LSA database or advertisements to show only the data from sources having the specified type.

Can also be used with other subset options ( ospf-area-id, link-state-id, router-id, and sequence-number) to further define the source of displayed information.

detail

Displays LSA details for the OSPF area(s) configured on the routing switch. The output can also be filtered by area (area-id), link-state-id, router-id, and sequence-number. Default: All OSPF areas configured on the routing switch.

**Example**

When you enter this command, the switch displays an output similar to the following for all configured areas:

```
show ip ospf link-state output
```

### OSPF Link State Database for Area 0.0.0.0

<table>
<thead>
<tr>
<th>LSA Type</th>
<th>Link State ID</th>
<th>Advertising</th>
<th>Age</th>
<th>Sequence #</th>
<th>Checksum</th>
</tr>
</thead>
<tbody>
<tr>
<td>Router</td>
<td>10.0.8.32</td>
<td>10.0.8.32</td>
<td>65</td>
<td>0x80000281</td>
<td>0x0000a7b6</td>
</tr>
<tr>
<td>Router</td>
<td>10.0.8.33</td>
<td>10.0.8.33</td>
<td>1638</td>
<td>0x80000005</td>
<td>0x0000a7c8</td>
</tr>
<tr>
<td>Network</td>
<td>10.3.2.37</td>
<td>10.0.8.37</td>
<td>1695</td>
<td>0x80000006</td>
<td>0x00000443</td>
</tr>
<tr>
<td>Summary</td>
<td>10.3.16.0</td>
<td>10.0.8.33</td>
<td>1638</td>
<td>0x80000007</td>
<td>0x0000c242</td>
</tr>
<tr>
<td>Summary</td>
<td>10.3.16.0</td>
<td>10.0.8.35</td>
<td>1316</td>
<td>0x80000008</td>
<td>0x0000aa58</td>
</tr>
<tr>
<td>Summary</td>
<td>10.3.17.0</td>
<td>10.0.8.33</td>
<td>1638</td>
<td>0x8000027b</td>
<td>0x0000becf</td>
</tr>
<tr>
<td>Summary</td>
<td>10.3.17.0</td>
<td>10.0.8.35</td>
<td>1316</td>
<td>0x80000008</td>
<td>0x0000a957</td>
</tr>
<tr>
<td>AsbSummary</td>
<td>10.0.8.36</td>
<td>10.0.8.33</td>
<td>1412</td>
<td>0x80000002</td>
<td>0x0000c2ba</td>
</tr>
</tbody>
</table>

### OSPF Link State Database for Area 10.3.16.0

<table>
<thead>
<tr>
<th>LSA Type</th>
<th>Link State ID</th>
<th>Advertising</th>
<th>Age</th>
<th>Sequence #</th>
<th>Checksum</th>
</tr>
</thead>
<tbody>
<tr>
<td>Router</td>
<td>10.0.8.33</td>
<td>10.0.8.33</td>
<td>1727</td>
<td>0x8000027e</td>
<td>0x0000d53c</td>
</tr>
<tr>
<td>Router</td>
<td>10.0.8.34</td>
<td>10.0.8.34</td>
<td>1420</td>
<td>0x80000283</td>
<td>0x0000de4f</td>
</tr>
<tr>
<td>Network</td>
<td>10.3.16.34</td>
<td>10.0.8.34</td>
<td>1735</td>
<td>0x80000005</td>
<td>0x00001465</td>
</tr>
</tbody>
</table>

The OSPF link-state display shows the following contents of the LSA database; one table for each area:

**Table 25: CLI display of OSPF link-state information**

<table>
<thead>
<tr>
<th>Field</th>
<th>Content</th>
</tr>
</thead>
<tbody>
<tr>
<td>LSA Type</td>
<td>The possible types are:</td>
</tr>
<tr>
<td></td>
<td>▪ Router</td>
</tr>
<tr>
<td></td>
<td>▪ Network</td>
</tr>
<tr>
<td></td>
<td>▪ Summary</td>
</tr>
<tr>
<td></td>
<td>▪ AsbSummary</td>
</tr>
</tbody>
</table>
### Field | Content
--- | ---
Link State ID | LSA ID for this LSA. The meaning depends on the LSA type.
Advertised Router ID | Router ID of the router that originated this LSA.
Age | Current age (in seconds) of this LSA.
Sequence # | Sequence number of the current instance of this LSA.
Chksum(Hex) | LSA checksum value.

There is a software limitation for sending LSAs from OSPFv3 database to SNMP/CLI. A single query can take 128 links at the maximum. Hence, only up to 128 link entries per router LSA is displayed.

**Output for show ip ospf link-state advertise**

```
switch(config)# show ip ospf link-state advertise

OSPF Link State Database for Area 0.0.0.0
Advertisements
-----------------------------------------------
000202010a000820a00082080000281a7b6005400000050a30e00ffff00030000001...
000202010a0008210a00082180000006a5c9002401000010a0008230a03112104000002
0001012010a0008230a0008238000015755d006c01000070a03060000ff000030000001...
000202020a0008220a0008228000027fd33d0054050000050a30e00ffff00030000002
0001012010a0008210a00082180000006a5c9002401000010a0008230a03112104000002
0001012010a0008230a0008238000015755d006c01000070a03060000ff000030000001...
000202020a0008210a0008220a0008228000027fd33d0054050000050a30e00ffff00030000001...
0001012010a0008220a00082280000027fd33d0054050000050a30e00ffff00030000001...
0001012010a0008230a0008238000015755d006c01000070a03060000ff000030000001...

OSPF Link State Database for Area 10.3.16.0
Advertisements
-----------------------------------------------
000202010a0008210a000821800000027fd33d0054050000050a30e00ffff00030000002
0001012010a0008220a0008228000027fd33d0054050000050a30e00ffff00030000002
0001012010a0008230a0008238000015755d006c01000070a03060000ff000030000002
```

**Output for show IP OSPF link-state detail for router**

This is an example of `show ip ospf link-state detail` output for a router.

```
switch(config)# show ip ospf link-state detail

OSPF Link State Database for Area 0.0.0.0

LSA Age : 35
LSA Type : 0x1 (Router)
Advertising Router : 2.2.2.3
Link State ID : 2.2.2.3
LSA Sequence : 0x80000007
LSA Checksum : 0xfd09
LSA Option Bits : E=1 MC=0 N/P=0 EA=0 DC=1
Router Capability Bits : B=0 E=1 V=0
Number of links : 1
Interface Type : 2 (Connected to Transit Network)
LSA Metric : 1
Link Data : 2.2.2.3
LSA ID : 2.2.2.3

Output for show IP OSPF link-state detail for a network
This is an example of show ip ospf link-state detail summary for LSA detailed output.

cswitch(config)# show ip ospf link-state detail

OSPF Link State Database for Area 0.0.0.0

| LSA Age | 19 |
| LSA Type | 0x2 (Network) |
| Advertising Router | 16.93.223.84 |
| Link State ID | 192.22.23.24 |
| LSA Sequence | 0x80000001 |
| LSA Checksum | 0x323e |
| LSA Option Bits | E=1 MC=0 N/P=0 EA=0 DC=1 |
| Network Mask | 255.255.255.0 |
| Attached Router ID | 2.2.2.3 |
| Attached Router ID | 192.93.226.105 |

Output for show IP OSPF link-state detail for summary of LSA detailed output
This is an example of show ip ospf link-state detail summary of LSA for AS Boundary Router.

cswitch(config)# show ip ospf link-state detail

OSPF Link State Database for Area 0.0.0.0

| LSA Age | 58 |
| LSA Type | 0x4 (AS Boundary) |
| Advertising Router | 16.93.226.105 |
| Link State ID | 2.2.2.3 |
| LSA Sequence | 0x80000001 |
| LSA Checksum | 0x4bc4 |
| LSA Option Bits | E=1 MC=0 N/P=0 EA=0 DC=1 |
| LSA Metric | 1 |

Output for show IP OSPF link-state detail for AS external LSA
This example shows show ip ospf link-state detail for an AS external LSA.

cswitch(config)# show ip ospf link-state detail

LSA Age : 971
LSA Type : 0x5 (AS External)
Advertising Router : 2.2.2.3
Link State ID : 55.5.5.0
LSA Sequence : 0x80000001
LSA Checksum : 0xe17c
LSA Option Bits : E=1 MC=0 N/P=0 EA=0 DC=0
LSA Metric : 10
Bit E : 0
Forwarding Address : 0.0.0.0

Output for show IP OSPF link-state detail for summary for NSSA
This example shows `show ip ospf link-state detail summary` for NSSA.

```
switch(config)# show ip ospf link-state detail

LSA Age : 86
LSA Type : 0x7 (NSSA)
Advertising Router : 16.93.226.105
Link State ID : 16.93.49.0
LSA Sequence : 0x80000003
LSA Checksum : 0x6c03
LSA Option Bits : E=1 MC=0 N/P=0 EA=0 DC=1
LSA Metric : 10
Network Mask : 255.255.255.0
Bit E : 0 (External Metric Type1)
Forwarding Address : 0.0.0.0
External Route Tag : 0
```

Viewing OSPF neighbor information

**Syntax:**

```
show ip ospf neighbor [ip-addr]
```

To display OSPF information for all neighbors, enter `show ip ospf neighbor` at any CLI level.

[ip-addr] can be specified to retrieve detailed information for the specific neighbor only. This is the IP address of the neighbor, not the router ID.

**Output for the show ip ospf neighbor command**

```
OSPF Neighbor Information

<table>
<thead>
<tr>
<th>Router ID</th>
<th>Pri</th>
<th>IP Address</th>
<th>NbIfState</th>
<th>State</th>
<th>Rxmt</th>
<th>QLen</th>
<th>Events</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0.8.34</td>
<td>1</td>
<td>10.3.18.34</td>
<td>DR</td>
<td>FULL</td>
<td>0</td>
<td>6</td>
<td></td>
<td>none</td>
</tr>
<tr>
<td>10.3.53.38</td>
<td>1</td>
<td>10.3.53.38</td>
<td>DR</td>
<td>FULL</td>
<td>0</td>
<td>6</td>
<td></td>
<td>none</td>
</tr>
</tbody>
</table>
```

This display shows the following information.

**Table 26: CLI display of OSPF neighbor information**

<table>
<thead>
<tr>
<th><strong>Field</strong></th>
<th><strong>Description</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>Router ID</td>
<td>The router ID of the neighbor.</td>
</tr>
<tr>
<td>Pri</td>
<td>The OSPF priority of the neighbor. The priority is used during election of the DR and BDR.</td>
</tr>
<tr>
<td>IP Address</td>
<td>The IP address of this routing switch's interface with the neighbor.</td>
</tr>
<tr>
<td>NbIfState</td>
<td>The neighbor interface state. The possible values are:</td>
</tr>
<tr>
<td></td>
<td>• DR</td>
</tr>
<tr>
<td></td>
<td>This neighbor is the elected designated router for the interface.</td>
</tr>
<tr>
<td></td>
<td>• BDR</td>
</tr>
<tr>
<td></td>
<td>This neighbor is the elected backup designated router for the interface.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>-------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>blank</td>
<td>This neighbor is neither the DR or the BDR for the interface.</td>
</tr>
<tr>
<td>State</td>
<td>The state of the conversation (the adjacency) between your routing switch and the neighbor. The possible values are:</td>
</tr>
<tr>
<td>INIT</td>
<td>A Hello packet has recently been seen from the neighbor. However, bidirectional communication has not yet been established with the neighbor. (The switch itself did not appear in the neighbor's hello packet.) All neighbors in this state (or higher) are listed in the hello packets sent from the associated interface.</td>
</tr>
<tr>
<td>2WAY</td>
<td>Communication between the two routers is bidirectional. This is the most advanced state before beginning adjacency establishment. The DR and BDR are selected from the set of neighbors in the 2Way state or greater.</td>
</tr>
<tr>
<td>EXSTART</td>
<td>The first step in creating an adjacency between the two neighboring routers. The goal of this step is to decide which router is the main router and to decide upon the initial database description (DD) sequence number. Neighbor conversations in this state or greater are called adjacencies.</td>
</tr>
<tr>
<td>EXCHANGE</td>
<td>The switch is describing its entire link state database by sending DD packets to the neighbor. Each DD packet has a DD sequence number and is explicitly acknowledged. Only one DD packet can be outstanding at any time. In this state, link-state request packets can also be sent asking for the neighbor's more recent advertisements. All adjacencies in exchange state or greater are used by the flooding procedure. In fact, these adjacencies are fully capable of transmitting and receiving all types of OSPF routing protocol packets.</td>
</tr>
<tr>
<td>LOADING</td>
<td>Link-state request packets are sent to the neighbor asking for the more recent advertisements that have been discovered (but not yet received) in the exchange state.</td>
</tr>
<tr>
<td>FULL</td>
<td>The neighboring routers are fully adjacent. These adjacencies will now appear in router links and network link advertisements.</td>
</tr>
</tbody>
</table>

| Rxmt QLen   | Remote transmit queue length—The number of LSAs that the routing switch has sent to this neighbor and for which the routing switch is awaiting acknowledgments.                                                   |
| Events      | The number of times the neighbor's state has changed.                                                                                                                                                     |
| Helper Status | Whether the neighboring router is helping the OSPF router. The possible values are:                                                        |
| Helper      | The neighbor is helping.                                                                                                                                                                                 |
| None        | The neighbor is not helping.                                                                                                                                                                             |
| Helper Age  | Amount of time the neighboring router is helping. This time can range from 1 to 1800 seconds with a default time of 120 seconds. Helper Age is 0 when the router is not helping.                               |

**Viewing OSPF redistribution information**
As described under Enabling route redistribution, you can configure the routing switch to redistribute connected, static, and RIP routes into OSPF. When you redistribute a route into OSPF, the routing switch can use OSPF to advertise the route to its OSPF neighbors.

To display the status of the OSPF redistribution, enter `show ip ospf redistribute` at any CLI context level:

**Example of output for show ip ospf redistribute**

```
switch# show ip ospf redistribute
OSPF redistributing
Route type Status
-------- --------
connected enabled
static enabled
rip enabled
```

The display shows whether redistribution of each of the route types, connected, static, and RIP is enabled.

**Viewing OSPF redistribution filter (restrict) information**

As described under Configuring external route redistribution in an OSPF domain (optional), you can configure the redistribution filters on the routing switch to restrict route redistribution by OSPF.

To display the status of the OSPF redistribution filters, enter `show ip ospf restrict` at any CLI context level.

**Example of output for show ip ospf restrict**

```
switch# show ip ospf restrict
OSPF restrict list

<table>
<thead>
<tr>
<th>IP Address</th>
<th>Mask</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0.8.0</td>
<td>255.255.248.0</td>
</tr>
<tr>
<td>15.0.0.0</td>
<td>255.0.0.0</td>
</tr>
</tbody>
</table>
```

This display shows the configured restrict entries.

**Viewing OSPF virtual neighbor information**

If virtual links are configured on the routing switch, you can display OSPF virtual neighbor information.

**Syntax:**

`show ip ospf virtual-neighbor [area area-id] [ip-address]`

**Output for the show ip ospf virtual-neighbor command**

```
OSPF Virtual Interface Neighbor Information

<table>
<thead>
<tr>
<th>Router ID</th>
<th>Area ID</th>
<th>State</th>
<th>IP Address</th>
<th>Events</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0.8.33</td>
<td>10.3.16.0</td>
<td>FULL</td>
<td>10.3.17.33</td>
<td>5</td>
</tr>
<tr>
<td>10.0.8.36</td>
<td>10.3.16.0</td>
<td>FULL</td>
<td>10.3.18.36</td>
<td>5</td>
</tr>
</tbody>
</table>
```

This display shows the following information.
Table 27: CLI display of OSPF virtual neighbor information

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Router ID</td>
<td>The router ID of this virtual neighbor (configured.)</td>
</tr>
<tr>
<td>Area ID</td>
<td>The area ID of the transit area for the virtual link to this neighbor (configured.)</td>
</tr>
<tr>
<td>State</td>
<td>The state of the adjacency with this virtual neighbor. The possible values are the same as the OSPF neighbor states. Virtual neighbors should never stay in the 2WAY state.</td>
</tr>
<tr>
<td>IP Address</td>
<td>IP address of the virtual neighbor that the routing switch is using to communicate to that virtual neighbor.</td>
</tr>
<tr>
<td>Events</td>
<td>The number of times the virtual neighbor's state has changed.</td>
</tr>
</tbody>
</table>

Notice from the syntax statement that `ip-address` can be specified to display detailed information for a particular virtual neighbor. If an `area-id` is specified, only virtual neighbors belonging to that area are shown.

Viewing OSPF virtual link information

**Syntax:**

```
show ip ospf virtual-link [[area area-id] | [ip-address]]
```

- `ip-address` Displays detailed information for a particular virtual neighbor.
- `area-id` Only virtual neighbors belonging to that area are shown.

**Output for the show ip ospf virtual-link command**

If virtual links are configured on a routing switch, you can display OSPF virtual link information by entering `show ip ospf virtual-link` at any CLI level.

```
switch# show ip ospf virtual-link

OSPF Virtual Interface Status

<table>
<thead>
<tr>
<th>Transit AreaID</th>
<th>Neighbor Router</th>
<th>Authentication</th>
<th>Interface State</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.3.16.0</td>
<td>10.0.8.33</td>
<td>none</td>
<td>P2P</td>
</tr>
<tr>
<td>10.3.16.0</td>
<td>10.0.8.36</td>
<td>none</td>
<td>P2P</td>
</tr>
</tbody>
</table>
```

This display shows the following information.

Table 28: CLI display of OSPF virtual link information

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Transit Area ID</td>
<td>Area ID of transit area for the virtual link.</td>
</tr>
</tbody>
</table>
### Field | Description
--- | ---
Neighbor Router | Router ID of the virtual neighbor.
Authentication | none or simple (same as for normal interface.)
Interface State | The state of the virtual link to the virtual neighbor. The possible values are:
  - **DOWN**
    - The routing switch has not yet found a route to the virtual neighbor.
  - **P2P (point-to-point)**
    - The routing switch has found a route to the virtual neighbor. Virtual links are "virtual" serial links, hence the point-to-point terminology.

Notice from the syntax statement that `ip-address` can be specified to display detailed information for a particular virtual neighbor. If an `area-id` is specified, only virtual neighbors belonging to that area are shown.

### Example
To get OSPF virtual link information for IP address 10.0.8.33, enter `show ip ospf virtual-link 10.0.8.33`. A display similar to the following is shown.

#### Output for the show ip ospf virtual-link ip-addr command

```
switch# show ip ospf virtual-link 10.0.8.33

OSPF Virtual Interface Status for interface 10.0.8.33
  Transit AreaID : 10.3.16.0
  Neighbor Router : 10.0.8.33

  Authentication : none           Chain   :
  Interface State : P2P            Transit Delay : 1
  Events           : 1               Rtr Interval : 5
  Dead Interval    : 40              Hello Interval : 10
```

In this display, these fields show the same type of information as described for the general OSPF virtual link display: Transit Area ID, Neighbor Router, Authentication, and Interface State. This display shows the following additional information:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Events</td>
<td>The number of times the virtual link interface state has changed.</td>
</tr>
<tr>
<td>Transit delay</td>
<td>The configured transit delay for the virtual link.</td>
</tr>
<tr>
<td>Rtr Interval</td>
<td>The configured retransmit interval for the virtual link.</td>
</tr>
<tr>
<td>Hello Interval</td>
<td>The configured hello interval for the virtual link.</td>
</tr>
<tr>
<td>Dead Interval</td>
<td>The configured router dead interval for the virtual link.</td>
</tr>
</tbody>
</table>

### Viewing OSPF SPF statistics
Displays the log used to record SPF calculations on an OSPF-enabled routing switch. The SPF algorithm recalculates the routes in an OSPF domain when a change in the area topology is received.

Syntax:
```
show ip ospf spf-log
```

This command output displays:

- The number of times that the SPF algorithm was executed for each OSPF area to which the routing switch is assigned. The event that resulted in the last ten executions of the SPF algorithm on the routing switch. Possible events (reasons) are as follows:
  - Re-init
    OSPF was enabled or disabled on the routing switch.
  - Router LS update
    A router (type 1) link-state advertisement was received.
  - Network LS update
    A network (type 2) link-state advertisement was received.
  - Generated RTR LSA
    A router (type 1) link-state advertisement was generated on the routing switch.
  - Generated NTW LSA
    A network (type 2) link-state advertisement was generated on the routing switch.

### Displaying OSPF SPF statistics

```
switch(ospf)# show ip ospf spf-log
OSPF SPF (SHORTEST PATH FIRST) LOG
Area : 0.0.0.100       - Number of times SPF executed : 12

<table>
<thead>
<tr>
<th>SPF Instance</th>
<th>Reason</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Router LS Update</td>
<td>0h:35m:44</td>
</tr>
<tr>
<td>2</td>
<td>Router LS Update</td>
<td>0h:36m:03</td>
</tr>
<tr>
<td>3</td>
<td>Generated RTR LSA</td>
<td>1h:04m:21</td>
</tr>
<tr>
<td>4</td>
<td>Generated NTW LSA</td>
<td>1h:28m:12</td>
</tr>
<tr>
<td>5</td>
<td>Network LS Update</td>
<td>2h:11m:05</td>
</tr>
<tr>
<td>6</td>
<td>Network LS Update</td>
<td>2h:54m:55</td>
</tr>
<tr>
<td>7</td>
<td>Generated RTR LSA</td>
<td>3h:01m:11</td>
</tr>
<tr>
<td>8</td>
<td>Router LS Update</td>
<td>3h:22m:39</td>
</tr>
<tr>
<td>9</td>
<td>Generated RTR LSA</td>
<td>4h:36m:22</td>
</tr>
<tr>
<td>10</td>
<td>Re-Init</td>
<td>4h:48m:54</td>
</tr>
</tbody>
</table>
```

| area [ area id | ID number or IP address of an area to which the switch is assigned, including the number of times the SPF algorithm was executed to recalculate OSPF routes in the area. |
| ip-address]   |                                                               |
| SPF           | Last ten instances in which the SPF algorithm was executed to recalculate an OSPF route in the area. |
| instances     |                                                               |
| Reason        | The event or reason why the SPF algorithm was executed.       |
| Time          | Time when the SPF computation began.                          |

### Displaying OSPF route information

Syntax:
```
show ip ospf
```
To display OSPF route and other OSPF configuration information, enter `show ip ospf` at any CLI level.

Output for show IP OSPF

```
switch# show ip ospf

OSPF Configuration Information

    OSPF protocol : enabled
    Router ID : 10.0.8.33

Currently defined areas:

<table>
<thead>
<tr>
<th>Area ID</th>
<th>Stub Type</th>
<th>Default Cost</th>
<th>Stub LSA Type</th>
<th>Stub Metric Type</th>
<th>SPF Runs</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>backbone</td>
<td>normal</td>
<td></td>
<td>don't send</td>
<td>ospf metric 1</td>
<td></td>
</tr>
<tr>
<td>10.3.16.0</td>
<td>normal</td>
<td></td>
<td>don't send</td>
<td>ospf metric 1</td>
<td></td>
</tr>
<tr>
<td>10.3.32.0</td>
<td>normal</td>
<td></td>
<td>don't send</td>
<td>ospf metric 1</td>
<td></td>
</tr>
</tbody>
</table>

Currently defined address ranges:

<table>
<thead>
<tr>
<th>Area ID</th>
<th>LSA Type</th>
<th>IP Network</th>
<th>Network Mask</th>
<th>Advertise Cost</th>
<th>Cost</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10.3.16.0</td>
<td>Summary</td>
<td>10.3.16.0</td>
<td>255.255.255.0</td>
<td>yes</td>
<td>1</td>
</tr>
</tbody>
</table>

OSPF interface configuration:

<table>
<thead>
<tr>
<th>IP Address</th>
<th>Area ID</th>
<th>Admin Status</th>
<th>Type</th>
<th>Authen Type</th>
<th>Cost</th>
<th>Pri</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10.3.2.35</td>
<td>backbone</td>
<td>enabled</td>
<td></td>
<td>BCAST</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>10.3.3.35</td>
<td>backbone</td>
<td>enabled</td>
<td></td>
<td>BCAST</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>10.3.16.35</td>
<td>10.3.16.0</td>
<td>enabled</td>
<td></td>
<td>BCAST</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>10.3.32.35</td>
<td>10.3.32.0</td>
<td>enabled</td>
<td></td>
<td>BCAST</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

OSPF configured interface timers:

<table>
<thead>
<tr>
<th>IP Address</th>
<th>Transit Delay</th>
<th>Retransmit Interval</th>
<th>Hello Interval</th>
<th>Dead Interval</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10.3.2.35</td>
<td>1</td>
<td>5</td>
<td>10</td>
<td>40</td>
</tr>
<tr>
<td>10.3.3.35</td>
<td>1</td>
<td>5</td>
<td>10</td>
<td>40</td>
</tr>
<tr>
<td>10.3.16.35</td>
<td>1</td>
<td>5</td>
<td>10</td>
<td>40</td>
</tr>
<tr>
<td>10.3.32.35</td>
<td>1</td>
<td>5</td>
<td>10</td>
<td>40</td>
</tr>
</tbody>
</table>

OSPF configured virtual interfaces:

<table>
<thead>
<tr>
<th>Area ID</th>
<th>Router ID</th>
<th>Authen Type</th>
<th>Xmit Delay</th>
<th>Rxmt Delay</th>
<th>Hello Interval</th>
<th>Dead Interval</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10.3.16.0</td>
<td>10.0.8.33</td>
<td>none</td>
<td>1</td>
<td>5</td>
<td>10</td>
<td>40</td>
</tr>
<tr>
<td>10.3.16.0</td>
<td>10.0.8.36</td>
<td>none</td>
<td>1</td>
<td>5</td>
<td>10</td>
<td>40</td>
</tr>
</tbody>
</table>

Table 32: CLI display of OSPF route and status information

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>OSPF protocol</td>
<td>enabled</td>
</tr>
<tr>
<td></td>
<td>or disabled</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Area ID</th>
<th>Router ID</th>
<th>Authen Type</th>
<th>Xmit Delay</th>
<th>Rxmt Delay</th>
<th>Hello Interval</th>
<th>Dead Interval</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.3.16.0</td>
<td>10.0.8.33</td>
<td>none</td>
<td>1</td>
<td>5</td>
<td>10</td>
<td>40</td>
</tr>
<tr>
<td>10.3.16.0</td>
<td>10.0.8.36</td>
<td>none</td>
<td>1</td>
<td>5</td>
<td>10</td>
<td>40</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Router ID</td>
<td>The router ID that this routing switch is currently using to identify itself.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Currently defined areas:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Area ID</td>
<td>The identifier for this area.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Type</td>
<td>The type of OSPF area (normal or stub.)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Stub Default Cost</td>
<td>The metric for any default route we injected into a stub area if the routing switch is an ABR for the area. This value applies only to stub areas.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Stub Summary LSA</td>
<td>send or don't send — indicates the state of the no-summary option for the stub area. The value indicates if the area is &quot;totally stubby&quot; (no summaries sent from other areas) or just &quot;stub&quot; (summaries sent.) Applies only to stub areas and takes effect only if the routing switch is the ABR for the area.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Stub Metric Type</td>
<td>This value is always ospf metric.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Currently defined address ranges:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Area ID</td>
<td>The area where the address range is configured.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LSA Type</td>
<td>This value is always Summary.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>IP Network</td>
<td>The address part of the address range specification.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Network Mask</td>
<td>The mask part of the address range specification.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Advertise</td>
<td>Whether advertising (yes) or suppressing (no) this address range.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Cost</td>
<td>The cost of the interface connection between one switch and another, which is determined by the bandwidth in mega bits per second. The OSPF protocol determines the interface connection cost of each neighbor and uses these costs to determine the best path to reach a destination. The cost can range from a minimum of 1 to a maximum of 10. The faster the connection, the lower the cost. For example, a fast Ethernet interface cost is 1 and a Ethernet interface cost is 10.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The remaining interface and virtual link information is the same as for the previously described OSPF show commands.

**Viewing OSPF traps enabled**
In the default configuration, OSPF traps are disabled. Use this command to view which OSPF traps have been enabled.

**Syntax:**
show ip ospf traps

Lists the OSPF traps currently enabled on the routing switch.
For more information, see [Changing OSPF trap generation choices](#).

### Debugging OSPF routing messages

**Syntax:**
debug ip ospf

Turns on the tracing of OSPF packets and displays OSPF routing messages.

### Enabling load sharing among next-hop routes

For more information, see [OSPF equal-cost multipath (ECMP) for different subnets available through the same next-hop routes](#).

**Syntax:**
ip load-sharing 2-4

When OSPF is enabled and multiple, equal-cost, next-hop routes are available for traffic destinations on different subnets, this feature, by default, enables load-sharing among up to four next-hop routes.

* 1 - 4 : Specifies the maximum number of equal-cost next-hop paths the router allows.

Default: 4; range: 2-4

The no form of the command disables this load-sharing so that only one route in a group of multiple, equal-cost, next-hop routes is used for traffic that could otherwise be load-shared across multiple routes.

For example, in [OSPF equal-cost multipath (ECMP) for different subnets available through the same next-hop routes](#), the next-hop routers "B", "C", and "D" are available for equal-cost load-sharing of eligible traffic. Disabling IP load-sharing means that router "A" selects only one next-hop router for traffic that is actually eligible for load-sharing through different next-hop routers.

Default: Enabled with four equal-cost, next-hop routes allowed.

---

**NOTE**

This command enables or disables load-sharing for both IPv4 (OSPFv2) and IPv6 (OSPFv3) operation.

For more information, see the IPv6 configuration guide for your switch.

In the default configuration, IP load-sharing is enabled by default. However, it has no effect unless IP routing and OSPF are enabled.

### Viewing the current IP load-sharing configuration

Use the `show running` command to view the currently active IP load-sharing configuration, and `show config` to view the IP load-sharing configuration in the startup-config file. (While in its default configuration, IP load-sharing does not appear in the command output.)

If IP load sharing is configured with non-default settings (disabled or configured for either two or three equal-cost next-hop paths), the current settings are displayed in the command output.
Overview of OSPF

OSPF is a link-state routing protocol applied to routers grouped into OSPF areas identified by the routing configuration on each routing switch. The protocol uses LSAs transmitted by each router to update neighboring routers regarding its interfaces and the routes available through those interfaces. Each routing switch in an area also maintains a link-state database (LSDB) that describes the area topology. (All routers in a given OSPF area have identical LSDBs.) The routing switches used to connect areas to each other flood summary link LSAs and external link LSAs to neighboring OSPF areas to update them regarding available routes. Through this means, each OSPF router determines the shortest path between itself and a desired destination router in the same OSPF domain (AS.) Routed traffic in an OSPF AS is classified as one of the following:

- Intra-area traffic
- Inter-area traffic
- External traffic

The switches support the following types of LSAs, which are described in RFCs 2328 and 3101:

**Table 33: OSPF LSA types**

<table>
<thead>
<tr>
<th>LSA type</th>
<th>LSA name</th>
<th>Use</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Router link</td>
<td>Describes the state of each interface on a router for a given area. Not propagated to backbone area.</td>
</tr>
<tr>
<td>2</td>
<td>Network link</td>
<td>Describes the OSPF routers in a given network. Not propagated to backbone area.</td>
</tr>
<tr>
<td>3</td>
<td>Summary link</td>
<td>Describes the route to networks in another OSPF area of the same AS. Propagated through backbone area to other areas.</td>
</tr>
<tr>
<td>4</td>
<td>Autonomous System (AS) summary link</td>
<td>Describes the route to an ASBR in an OSPF normal or backbone area of the same AS. Propagated through backbone area to other areas.</td>
</tr>
<tr>
<td>LSA type</td>
<td>LSA name</td>
<td>Use</td>
</tr>
<tr>
<td>---------</td>
<td>-----------------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>5</td>
<td>AS external link</td>
<td>Describes the route to a destination in another AS (external route.) Originated by ASBR in normal or backbone areas of an AS and propagates through backbone area to other normal areas. For injection into an NSSA, ABR converts type-5 LSAs to a type-7 LSA advertising the default route (0.0.0.0/0.)</td>
</tr>
<tr>
<td>7</td>
<td>AS external link in an NSSA</td>
<td>Describes the route to a destination in another AS (external route.) Originated by ASBR in NSSA. ABR converts type-7 LSAs to type-5 LSAs for injection into the backbone area.</td>
</tr>
</tbody>
</table>

**OSPF router types**

**Interior routers**

This type of OSPF router belongs to only one area. Interior routers flood type-1 LSAs to all routers in the same area and maintain identical LSDBs. In the following example, the routers R1, R3, R4, and R6 are all interior routers because all of their links are to other routers in the same area.

*Example of interior routers*

```
\begin{center}
\includegraphics[width=0.5\textwidth]{interior_router_diagram.png}
\end{center}
```

**Area border routers (ABRs)**

This type of OSPF router has membership in multiple areas. ABRs are used to connect the various areas in an AS to the backbone area for that AS. Multiple ABRs can be used to connect a given area to the backbone, and a given ABR can belong to multiple areas other than the backbone.

An ABR maintains a separate LSDB for each area to which it belongs. (All routers within the same area have identical LSDBs.) The ABR is responsible for flooding summary LSAs between its border areas. You can reduce summary LSA flooding by configuring area ranges. An area range enables you to assign an aggregate address to a range of IP addresses. This aggregate address is advertised instead of all the individual addresses it represents. You can assign up to eight ranges in an OSPF area. In the following example, routers R2 and R5 are ABRs because they both have membership in more than one area.
**Example of deploying ABRs to connect areas to the backbone**

![Diagram showing Aruba 3810M/5800R ABRs connecting areas to the backbone]

**Autonomous system boundary router (ASBR)**

This type of OSPF router runs multiple interior gateway protocols and serves as a gateway to other autonomous systems operating with interior gateway protocols. The ASBR imports and translates different protocol routes into OSPF through redistribution. ASBRs can be used in backbone areas, normal areas, and NSSAs, but not in stub areas. For more details on redistribution and configuration examples, see [Enabling route redistribution](#).

**Designated routers (DRs)**

In an OSPF network having two or more routers, one router is elected to serve as the DR and another router to act as the BDR. All other routers in the area forward their routing information to the DR and BDR, and the DR forwards this information to all of the routers in the network. This minimizes the amount of repetitive information that is forwarded on the network by eliminating the need for each individual router in the area to forward its routing information to all other routers in the network. If the area includes multiple networks, each network elects its own DR and BDR.

In an OSPF network with no DR and no BDR, the neighboring router with the highest priority is elected the DR, and the router with the next highest priority is elected the BDR. If the DR goes off-line, the BDR automatically becomes the DR, and the router with the next highest priority then becomes the new BDR. If multiple routing switches on the same OSPF network are declaring themselves DRs, both priority and router ID are used to select the DR and BDRs.

Priority is configurable by using the `vlan vid ip ospf priority 0-255` command at the interface level. You can use this parameter to help bias one router as the DR. For more information, see [Changing priority per-interface](#). If two neighbors share the same priority, the router with the highest router ID is designated the DR. The router with the next highest router ID is designated the BDR.

For example, in [Example of DRs in an OSPF area](#), the DR and BDR for 10.10.10.0 network in area 5 are determined as follows:

<table>
<thead>
<tr>
<th>Router</th>
<th>Priority</th>
<th>DR/BDR for 10.10.10.0 network</th>
</tr>
</thead>
<tbody>
<tr>
<td>Router A</td>
<td>0</td>
<td>Cannot become a DR or BDR</td>
</tr>
<tr>
<td>Router B</td>
<td>1</td>
<td>DR for the 10.10.10.0 network</td>
</tr>
<tr>
<td>Router C</td>
<td>2</td>
<td>BDR for the 10.10.10.0 network</td>
</tr>
<tr>
<td>Router D</td>
<td>Priority: 3</td>
<td>Cannot become a DR or BDR</td>
</tr>
<tr>
<td>---------</td>
<td>-------------</td>
<td>--------------------------</td>
</tr>
<tr>
<td>Router E</td>
<td>Priority: 4</td>
<td>Becomes the new BDR if router B becomes unavailable and router C becomes the new DR</td>
</tr>
</tbody>
</table>

**Figure 9** *Example of DRs in an OSPF area*

<table>
<thead>
<tr>
<th>Area 0 (Backbone)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Router &quot;X&quot;</td>
</tr>
<tr>
<td>ID: 99.101.1.2/24</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Area 5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Router &quot;A&quot;</td>
</tr>
<tr>
<td>ID: 10.10.10.1/24</td>
</tr>
</tbody>
</table>

| Router "E" |
| Priority: 3 |
| ID: 10.10.10.4/24 |

| Router "D" |
| Priority: 0 |
| ID: 10.10.10.5/24 |

| Router "B" |
| Priority: 1 |
| ID: 10.10.10.2/24 |

| Router "C" |
| Priority: 2 |
| ID: 10.10.10.3/24 |

To learn the router priority on an interface, use the `show ip ospf interface` command and check the Pri setting under OSPF interface configuration.

---

**NOTE**

By default, the router ID is typically the lowest-numbered IP address or the lowest-numbered (user-configured) loopback interface configured on the device.

---

If multiple networks exist in the same OSPF area, the recommended approach is to ensure that each network uses a different router as its DR. Otherwise, if a router is a DR for more than one network, latency in the router could increase because of the increased traffic load resulting from multiple DR assignments.

---

When only one router on an OSPF network claims the DR role despite neighboring routers with higher priorities or router IDs, this router remains the DR. This is also true for BDRs.

The DR and BDR election process is performed when one of the following events occurs:

- Interface is in a waiting state and the wait time expires
- Interface is in a waiting state and a hello packet is received that addresses the BDR
- Change in the neighbor state occurs, such as:
  - Neighbor state transitions from 2 or higher
  - Communication to a neighbor is lost
  - Neighbor declares itself to be the DR or BDR for the first time

**OSPF area types**

OSPF is built upon a hierarchy of network areas. All areas for a given OSPF domain reside in the same AS. An AS is defined as a number of contiguous networks, all of which share the same interior gateway routing protocol. An AS can be divided into multiple areas. Each area represents a collection of contiguous networks and hosts, and the topology of a given area is not known by the internal routers in any other area. Areas define the boundaries to which types 1 and 2 LSAs are broadcast, which limits the amount of LSA flooding that occurs within the AS and also helps to control the size of the LSDBs maintained in OSPF routers. An area is represented in OSPF by either an IP address or a number. Area types include:

<table>
<thead>
<tr>
<th>Backbone</th>
<th>Not-so-stubby (NSSA)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Normal</td>
<td>Stub</td>
</tr>
</tbody>
</table>

All areas in an AS must connect with the backbone through one or more ABRs. If a normal area is not directly connected to the backbone area, it must be configured with a virtual link to an ABR that is directly connected to the backbone. The remaining area types do not allow virtual link connections to the backbone area.

*Example of an AS with multiple areas and external routes*

![OSPF Area Diagram](image)

**Backbone area**

Every AS must have one (and only one) backbone area (identified as area 0 or 0.0.0.0.) The ABRs of all other areas in the same AS connect to the backbone area, either physically through an ABR or through a configured, virtual link. The backbone is a transit area that carries the type-3 summary LSAs, type-5 AS external link LSAs and routed traffic between non-backbone areas, as well as the type-1 and type-2 LSAs and routed traffic internal to the area. ASBRs are allowed in backbone areas.

**Normal area**
This area connects to the AS backbone area through one or more ABRs (physically or through a virtual link) and supports type-3 summary LSAs and type-5 external link LSAs to and from the backbone area. ASBRs are allowed in normal areas.

**Not-so-stubby-area (NSSA)**

This area is available and connects to the backbone area through one or more ABRs. NSSAs are intended for use where an ASBR exists in an area where you want to control the following:

- Advertising the ASBR's external route paths to the backbone area
- Advertising the NSSA’s summary routes to the backbone area
- Allowing LSAs from the backbone area to advertise in the NSSA:
  - Summary routes (type-3 LSAs) from other areas
  - External routes (type-5 LSAs) from other areas as a default external route (type-7 LSAs)

In the above operation, the ABR in the NSSA injects external routes as type 7 LSAs. (Type 5 LSAs are not allowed in an NSSA.) The ABR connecting the NSSA to the backbone converts the type 7 LSAs to type 5 LSAs and injects them into the backbone area for propagation to networks in the backbone and to any normal areas configured in the AS. The ABR also injects type-3 summary LSAs:

- From the NSSA into the backbone area
- From the backbone into the NSSA

If the ABR detects type-5 external LSAs on the backbone, it injects a corresponding type-7 LSA default route (0.0.0.0/0) into the NSSA

You can also configure the NSSA ABR to do the following:

- Suppress advertising some or all of the area’s summarized internal or external routes into the backbone area.
  - See [Configuring ranges on an ABR to reduce advertising to the backbone](#).
- Replace all type-3 summary routes and the type-7 default route with the type-3 default summary route (0.0.0.0/0.)

**Stub area**

This area connects to the AS backbone through one or more ABRs. It does not allow an internal ASBR, and does not allow external (type 5) LSAs. A stub area supports these actions:

- Advertise the area’s summary routes to the backbone area.
- Advertise summary routes from other areas.
- Use the default summary (type-3) route to advertise both of the following:
  - Summary routes to other areas in the AS
  - External routes to other ASs

You can configure the stub area ABR to do the following:

- Suppress advertising some or all of the area’s summarized internal routes into the backbone area.
- Suppress LSA traffic from other areas in the AS by replacing type-3 summary LSAs and the default external route from the backbone area with the default summary route (0.0.0.0/0.)

Virtual links are not allowed for stub areas.
OSPF RFC compliance

The OSPF features covered in this guide comply with the following:

- RFC 2328 OSPF version 2
- RFC 3101 OSPF NSSA option
- RFC 1583 (Enabled in the default OSPF configuration. See the following Note.)
- RFC 4750 MIB variables.

If all of the routers in your OSPF domain support RFC 2178, RFC 2328, or later, you should disable RFC 1583 compatibility on all routers in the domain. See Changing the RFC 1583 OSPF compliance setting.

Reducing AS external LSAs and Type-3 summary LSAs

An OSPF ASBR uses AS external LSAs to originate advertisements of a route to another routing domain, such as an RIP domain. These advertisements are

- Flooded in the area in which the ASBR operates
- Injected into the backbone area and then propagated to any other OSPF areas (except stub areas) within the local OSPF AS. If the AS includes an NSSA, there are two additional options:
  - If the NSSA includes an ASBR, you can suppress advertising some or all of its summarized external routes into the backbone area.
  - Replace all type-3 summary LSAs and the default external route from the backbone area with the default summary route (0.0.0.0/0).

In some cases, multiple ASBRs in an AS can originate equivalent external LSAs. The LSAs are equivalent when they have the same cost, the same next hop, and the same destination. In such cases, the switch optimizes OSPF by eliminating duplicate AS external LSAs. That is, the ASBR with the highest router ID floods the AS external LSAs for the external domain into the OSPF AS, while the other ASBRs flush the equivalent AS external LSAs from their databases. As a result, the overall volume of route advertisement traffic within the AS is reduced and the switches that flush the duplicate AS external LSAs have more memory for other OSPF data.

This enhancement implements the portion of RFC 2328 that describes AS external LSA reduction. This enhancement is enabled by default, requires no configuration, and cannot be disabled.

Algorithm for AS external LSA reduction

The AS external LSA reduction feature behavior changes under the following conditions:

- There is one ASBR advertising (originating) a route to the external destination, but one of the following happens:
  - A second ASBR comes on-line.
  - A second ASBR that is already on-line begins advertising an equivalent route to the same destination.
    In either of these cases, the switch with the higher router ID floods the AS external LSAs and the other switch flushes its equivalent AS external LSAs.
- One of the ASBRs starts advertising a route that is no longer equivalent to the route the other ASBR is advertising. In this case, the ASBRs each flood AS external LSAs. Since the LSAs either no longer have the same cost or no longer have the same next-hop router, the LSAs are no longer equivalent, and the LSA reduction feature no longer applies.
The ASBR with the higher router ID becomes unavailable or is reconfigured so that it is no longer an ASBR. In this case, the other ASBR floods the AS external LSAs.

**Replacing type-3 summary LSAs and type-7 default external LSAs with a type-3 default route LSA**

By default, a routing switch operating as an ABR for a stub area or NSSA injects non-default, summary routes (LSA type 3) into the stub areas and NSSAs. For NSSAs, the routing switch also injects a type-7 default external route. You can further reduce LSA traffic into these areas by using `no-summary`. This command option configures the routing switch to:

- Replace type-3 summary LSA injection into a stub area or NSSA with a type-3 default summary route (0.0.0.0/0).
- Disable injection of the type-7 default external route into an NSSA.

You can enable this behavior when you first configure the stub area or NSSA, or at a later time. For the full command to use, see [Configuring a stub or NSSA area](#).

The `no-summary` command does not affect intra-area advertisements, meaning the switch still accepts summary LSAs from OSPF neighbors within its area and floods them to other neighbors. The switch can form adjacencies with other routers regardless of whether summarization is enabled or disabled for areas on each switch.

When you use `no-summary`, the change takes effect immediately. If you apply the option to a previously configured area, the switch flushes all of the summary LSAs it has generated (as an ABR) from the area.

To implement the above operation for a stub area or NSSA, enter a command such as the following:

```
switch(ospf)# area 40 stub 3 no-summary
```

**Equal cost multi-path routing (ECMP)**

The ECMP feature allows OSPF to add routes with multiple next-hop addresses and with equal costs to a given destination in the forwarding information base (FIB) on the routing switch. For example, if you display the IP route table by entering the `show ip route` command, multiple next-hop routers are listed for the same destination network (21.0.9.0/24) as shown in following example.

**Example of show ip route command output with multiple next-hop routes**

```
switch show ip route

  IP Route Entries

  Destination        Gateway           VLAN Type Sub-Type Metric Dist.
  ----------------- ----------- -------------- ------- ------ ------- -----
  1.0.0.0/8          10.0.8.1         1 static      1        1      1
  10.0.8.0/21        DEFAULT_VLAN     1 connected   1        0      0
  12.0.9.0/24        VLAN3            3 connected   1        0      0
  15.0.0.0/8         10.0.8.1         1 static      1        1      1
  21.0.9.0/24        162.130.101.2   2 ospf        IntraArea 2    110
  21.0.9.0/24        162.130.101.3   2 ospf        IntraArea 2    110
  21.0.9.0/24        162.130.101.4   2 ospf        IntraArea 2    110
```

This feature applies only when the switch is configured as an ABR for a stub area or NSSA. To completely prevent summary LSAs from injection into the area, use `no-summary` to disable the summary LSAs on each OSPF router that is an ABR for the area.
For a given destination network in an OSPF domain, multiple ECMP next-hop routes can be one of the following types:

- Intra-area (routes to the destination in the same OSPF area)
- Inter-area (routes to the destination through another OSPF area)
- External (routes to the destination through another AS)

Multiple ECMP next-hop routes cannot be a mixture of intra-area, inter-area, and external routes. For example, in Equal cost multi-path routing (ECMP), the multiple next-hop routes to network 21.0.9.0/24 are all intra-area.

Also, according to the distributed algorithm used in the selection of ECMP next-hop routes:

- Intra-area routes are preferred to inter-area routes.
- Inter-area routes are preferred to external routes through a neighboring AS.

In addition, ECMP ensures that all traffic forwarded to a given host address follows the same path, which is selected from the possible next-hop routes.

For example, in the following figure, the ECMP inter-area routes to destination network 10.10.10.0/24 consist of the following next-hop gateway addresses: 12.0.9.2, 13.0.9.3, and 14.0.9.4.

**Example of OSPF ECMP multiple next-hop routing (inter-area)**

However, the forwarding software distributes traffic across the three possible next-hop routes in such a way that all traffic for a specific host is sent to the same next-hop router.

As shown in the following figure, one possible distribution of traffic to host devices is:

- Traffic to host 10.10.0.1 passes through next-hop router 12.0.9.2.
- Traffic to host 10.10.0.2 passes through next-hop router 13.0.9.3.
- Traffic to host 10.10.0.3 passes through next-hop router 12.0.9.2.
- Traffic to host 10.10.0.4 passes through next-hop router 14.0.9.4.

*Example of traffic distribution on ECMP next-hop routers*

<table>
<thead>
<tr>
<th>IP packet destination</th>
<th>Next hop used</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.10.0.1</td>
<td>12.0.9.2</td>
</tr>
<tr>
<td>10.10.0.2</td>
<td>13.0.9.3</td>
</tr>
<tr>
<td>10.10.0.3</td>
<td>12.0.9.2</td>
</tr>
<tr>
<td>10.10.0.4</td>
<td>14.0.9.4</td>
</tr>
</tbody>
</table>

**Dynamic OSPF activation and configuration**

OSPF automatically activates when enabled with `router ospf`. All configuration commands affecting OSPF (except reconfiguring the router ID) are dynamically implemented and can be used without restarting OSPF routing.

*NOTE* OSPF is automatically enabled without a system reset.

**General configuration steps for OSPF**

To begin using OSPF on the routing switch:

**Procedure**

1. In the global config context, use `ip routing` to enable routing.
2. Execute `router ospf` to place the routing switch in the `ospf` context and to enable OSPF routing (page A-21.)
3. Change the OSPF RFC 1583 compliance, if needed.
4. Use `area` to assign the areas to which the routing switch will be attached (page A-25.)
5. Assign interfaces to the configured areas per-VLAN or per-subnet by moving to each VLAN context and using one of the following commands:
   a. `ip ospf area ospf-area-id`
      Assigns all interfaces in the VLAN to the same area. Use this option when there is only one IP address configured on the VLAN or you want all subnets in the VLAN to belong to the same OSPF area.
   b. `ip ospf ip-address area ospf-area-id`
      Assigns an individual subnet to the specified area.
6. Optional: Assign loopback interfaces to OSPF areas by using the `ip ospf area` command at the loopback interface configuration level.
7. Optional: On each routing switch used as an ASBR in your OSPF domain, configure redistribution to enable importing the routes you want to make available in the domain.
   a. On an ASBR in a backbone, normal, or NSSA area where you want to import external routes, configure redistribution filters to define the external routes you do not want imported.
   b. Enable redistribution.

8. Optional: Use administrative distance to influence route choices.


10. Optional: Reconfigure default parameters in the interface context, if needed. Includes cost, dead-interval, hello-interval, priority, and others.


**Configuration rules**

- If the switch is to operate as an ASBR, you must enable redistribution (step 7 in General configuration steps for OSPF).

  When you do that, ASBR capability is automatically enabled. For this reason, you should first configure redistribution filters on the ASBR. Otherwise, all possible external routes will be allowed to flood the domain. (See Configuring external route redistribution in an OSPF domain (optional).)

- Each VLAN interface on which you want OSPF to run must be assigned to one of the defined areas. When a VLAN interface is assigned to an area, the IP address is automatically included in the assignment. To include additional addresses, you must enable OSPF on them separately, or use the "all" option in the assignment.

**OSPF global and interface settings**

When first enabling OSPF, you may want to consider configuring ranges and restricting redistribution (if an ASBR is used) to avoid unwanted advertisements of external routes. You may also want to enable the OSPF trap and authentication features to enhance troubleshooting and security. However, Hewlett Packard Enterprise generally recommends that the remaining parameters with non-null default settings be left as-is until you have the opportunity to assess OSPF operation and determine whether any adjustments to non-default settings is warranted.

---

Set global level parameters in the ospf context of the CLI. To access this context level, ensure that routing is enabled, then execute `router ospf` at the global CONFIG level. For example:

```
switch (config)# router ospf
switch (ospf)#
```

Use the VLAN interface context to set interface level OSPF parameters for the desired VLAN. To access this context level, use `vlan vid` either to move to the VLAN context level or to specify that context from the global config level. For example, both of the following command sets achieve the same result:

```
switch(config)# vlan 20
switch(vlan-20)# cost 15
```

```
switch(config)# vlan 20 cost 15
```
Changing the RFC 1583 OSPF compliance setting

For more information, see Changing the RFC 1583 OSPF compliance setting.

Syntax:

rfc1583-compatibility

Executed at the global configuration level to toggle routing switch operation compliance between RFC 1583 and RFC 2328.

rfc1583-compatibility

Configures the routing switch for external route preference rules compliant with RFC 1583.

no rfc1583-compatibility

Configures the routing switch for external route preference rules compliant with RFC 2328.

Default: Compliance enabled

Example

To disable RFC 1583 compatibility on a routing switch in an OSPF domain where RFC 2178 and RFC 2328 are universally supported:

```plaintext
switch(config)# router ospf
switch(ospf)# no rfc1583-compatibility
```

Assigning the routing switch to OSPF areas

For more information, see Assigning the routing switch to OSPF areas.

Configuring for external route redistribution in an OSPF domain

Configuring route redistribution for OSPF establishes the routing switch as an ASBR (residing in a backbone, normal, or NSSA) for importing and translating different protocol routes from other IGP domains into an OSPF domain. The switches support redistribution for static routes, RIP routes, and directly connected routes from RIP domains into OSPF domains. When you configure redistribution for OSPF, you can specify that static, connected, or RIP routes external to the OSPF domain are imported as OSPF routes. (Likewise, RIP
redistribution supports the import of static, connected, and OSPF routes into RIP routes.) The steps for configuring external route redistribution to support ASBR operation include the following:

1. Configure redistribution filters to exclude external routes that you do not want redistributed in your OSPF domain.

2. Enable route redistribution.

3. Modify the default metric for redistribution (optional.)

4. Modify the redistribution metric type (optional.)

5. Change the administrative distance setting (optional.)

---

**Do not enable redistribution until you have used restrict to configure the redistribution filters. Otherwise, your network might become overloaded with routes that you did not intend to redistribute.**

---

### Configuring ranges on an ABR to reduce advertising to the backbone

**Syntax:**

```
area [ospf-area-id] [backbone] range [ip-addr/mask-length] [no-advertise] [type summary [cost 1-16777215] [nssa] [cost 1-16777215]]
area ospf-area-id range ip-addr/mask-length [no-advertise] [type summary [cost 1-16777215] [nssa]]
```

Use this command on a routing switch intended to operate as an ABR for the specified area to do either of the following:

- Simultaneously create the area and corresponding range setting for routes to summarize or block.
- For an existing area, specify a range setting for routes to summarize or block.

**Options**

ospf-area-id

Same area ID as in Configuring an OSPF backbone or normal area, except you cannot assign a backbone area number (0 or 0.0.0.0) to a stub or NSSA area.

range ip-addr/mask-length

Defines the range of route advertisements to either summarize for injection into the backbone area or to prevent from being injected into the backbone area. The ip-addr value specifies the IP address portion of the range, and mask-length specifies the leftmost significant bits in the address. The ABR for the specified area compares the IP address of each outbound route advertisement with the address and significant bits in the mask to determine which routes to select for either summarizing or blocking. For example, a range of 10.10.32.1/14 specifies all routes in the range of 10.10.32.1 - 10.10.35.254.

no advertise

Use this keyword only if you want to configure the ABR to prevent advertisement to the backbone of a specified range of routes. (This has the effect of "hiding" the specified range from the backbone area.) If you do not use this option, the ABR advertises the specified range of routes according to the type summary | nssa selection described below.

[type summary [cost 1-16777215] [nssa]]

Configures the type of route summaries to advertise or block.
If the option `type` is not used in the command, the ABR defaults this setting to summary.

The option `type summary [cost 1-16777215]` specifies internal routes in the configured range of RAs and the user-configured cost for an area summary range. If cost is specified, the range will advertise the specified cost instead of the calculated cost.

If `no advertise` is used in the command, the ABR prevents the selected internal routes from being summarized in a type-3 LSA and advertised to the backbone.

If `no-advertise` is not used in the command, the selected routes are summarized to the backbone in a type-3 LSA.

Option `nssa` specifies external routes (type-7 LSAs) in the configured range of route advertisements. If `no-advertise` (above) is used in the command, the ABR prevents the selected external routes from being summarized in a type-5 LSA and advertised to the backbone. (Configure this option where an ABR for an NSSA advertises external routes that you do not want propagated to the backbone.)

If `no-advertise` is not used in the command, the selected routes learned from type-7 LSAs in the area are summarized to the backbone in a type-5 LSA. `[cost 1-16777215]` User configured cost for an NSSA summary range. If cost is not configured, the ABR will use the algorithm defined in RFC 3101 to compute the cost and metric-type of the summarized route. If cost is specified, then the range will advertise the specified cost as the cost of the summarized route.

**Influencing route choices by changing the administrative distance default (optional)**

The administrative distance value can be left in its default configuration setting unless a change is needed to improve OSPF performance for a specific network configuration.

The switch can learn about networks from various protocols, including RIP and OSPF. Consequently, the routes to a network may differ depending on the protocol from which the routes were learned. For the switches, the administrative distance for OSPF routes is set at 110 for all route types (external, inter-area, and intra-area.)

The switch selects one route over another based on the source of the route information. To do so, the switch can use the administrative distances assigned to the sources to influence route choices. You can change the distance settings in the OSPF global context to enable preference of one route type over another.

**Adjusting performance by changing the VLAN or subnet interface settings**

Optional: A setting described in this section can be configured with the same value across all subnets in a VLAN or be configured on a per-interface basis with different values.

---

**NOTE**

Most of the OSPF interface parameters also apply to virtual link configurations. However, when used on a virtual link configuration, the OSPF context requirement is different and the parameters are applied only to the interfaces included in the virtual link. See [Changing the dead interval on a virtual link](#).

---

**Configuring OSPF interface authentication**

Optional: For more information, see [Configuring OSPF interface authentication](#).

**Configuring an ABR to use a virtual link to the backbone**
All ABRs must have either a direct, physical or indirect, virtual link to the OSPF backbone area (0.0.0.0 or 0.) If an ABR does not have a physical link to the area backbone, the ABR can use a virtual link to provide a logical connection to another ABR having a direct physical connection to the area backbone. Both ABRs must belong to the same area, and this area becomes a transit area for traffic to and from the indirectly connected ABR.

A backbone area can be purely virtual with no physical backbone links. Also, virtual links can be "daisy chained." If so, the virtual link may not have one end physically connected to the backbone.

Because both ABRs in a virtual link connection are in the same OSPF area, they use the same transit area ID. This setting is automatically determined by the ABRs and should match the area ID value configured on both ABRs in the virtual link.

The ABRs in a virtual link connection also identify each other with a neighbor router setting:

- On the ABR having the direct connection to the backbone area, the neighbor router is the IP address of the router interface needing a logical connection to the backbone.
- On the opposite ABR (the one needing a logical connection to the backbone), the neighbor router is the IP address of the ABR that is directly connected to the backbone.

By default, the router ID is the lowest numbered IP address or (user-configured) loopback interface configured on the device.

When you establish an area virtual link, you must configure it on both of the ABRs (both ends of the virtual link.)

Adjusting virtual link performance by changing the interface settings

Optional: The OSPF interface parameters for this process are automatically set to their default values for virtual links. No change to the defaults is usually required unless needed for specific network conditions. These parameters are a subset of the parameters described under Adjusting performance by changing the VLAN or subnet interface settings. (The cost and priority settings are not configurable for a virtual link, and the commands for reconfiguring the settings are accessed in the router OSPF context instead of the VLAN context.)

The parameter settings for virtual links must be the same on the ABRs at both ends of a given link.

Configuring OSPF authentication on a virtual link

For more information, see Configuring OSPF authentication on a virtual link.

About OSPF passive

OSPF sends LSAs to all other routers in the same AS. To limit the flooding of LSAs throughout the AS, you can configure OSPF to be passive. OSPF does not run in the AS, but it does advertise the interface as a stub link into OSPF. Routing updates are accepted by a passive interface, but not sent out.

There is a limit of 512 total active and passive interfaces, but only a total of 128 can be active interfaces.

About configuring shortest path first (SPF) scheduling
SPF scheduling (throttling) can be configured in intervals of seconds to potentially delay SPF calculations when the network is unstable or there is a change in topology. It provides a granularity of one to four seconds between SPF calculations as opposed to the current default of five seconds.

The interval for the SPF calculations is dynamically chosen, based on the frequency of topology changes in the network. The chosen interval is within user-specified ranges of values. When the network topology is unstable, SPF throttling calculates SPF scheduling intervals that are longer, until the topology is again stable.

It is guaranteed that no SPF will be calculated within the SPF currently in effect, however, it is not guaranteed that the SPF will be calculated at the exact expiration of the timer if there have been updates. The timer may be delayed due to system constraints.

**Graceful shutdown of OSPF routing**

OSPF routing can be gracefully shut down on switches without losing packets that are in transit. OSPF neighbors are informed that the router should not be used for forwarding traffic, which allows for maintenance on the switch without interrupting traffic in the network. There is no effect on the saved switch configuration.

Prior to a switch shutdown, the CLI/SNMP `reload` command or the CLI `boot` command is executed to initiate the sending of OSPF "empty hello list" messages on the interfaces that are part of the OSPF routing configuration. After a small delay (approximately 2 seconds) that allows the messages to be transmitted on all applicable interfaces, the `boot` or `reload` command continues.

**Modules operating in nonstop mode**

When a switch is in standalone mode and OSPF routing is enabled, the "empty hello list" is transmitted whenever the `boot` or `reload` commands are executed.

When the switch is operating in nonstop switching mode (redundant) and a single module is being reloaded or booted, the standby module will notify neighboring switches of the management module failover. If the failover fails, the "empty hello list" is transmitted before the switch is rebooted.

When a switch is operating with multiple management modules in warm standby mode, the "empty hello list" is sent when a `reload` or `boot` command is executed. The standby management module sends out OSPF hello packets after becoming the active management module.

**OSPF equal-cost multipath (ECMP) for different subnets available through the same next-hop routes**

The switches support optional load-sharing across redundant links where the network offers two, three, or four equal-cost next-hop routes for traffic to different subnets. (All traffic for different hosts in the same subnet goes through the same next-hop router.)

For example, in the OSPF network shown in the following figure, IP load-sharing is enabled on router "A". In this case, OSPF calculates three equal-cost next-hop routes for each of the subnets and then distributes per-subnet route assignments across these three routes.

*Example of load-sharing traffic to different subnets through equal-cost next-hop routers*
Example of a routing table for the network in the preceding figure.

<table>
<thead>
<tr>
<th>Destination subnet</th>
<th>Router &quot;A&quot; next hop</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.1.0.0/16</td>
<td>Router &quot;C&quot;</td>
</tr>
<tr>
<td>10.2.0.0/16</td>
<td>Router &quot;D&quot;</td>
</tr>
<tr>
<td>10.3.0.0/16</td>
<td>Router &quot;B&quot;</td>
</tr>
<tr>
<td>10.32.0.0/16</td>
<td>Router &quot;B&quot;</td>
</tr>
<tr>
<td>10.42.0.0/16</td>
<td>Router &quot;D&quot;</td>
</tr>
</tbody>
</table>

IP load-sharing does not affect routed traffic to different hosts on the same subnet. That is, all traffic for different hosts on the same subnet will go through the same next-hop router. For example, if subnet 10.32.0.0 includes two servers at 10.32.0.11 and 10.32.0.22, all traffic from router "A" to these servers will go through router "B".

Global OSPF Cost Setting

The OSPF cost global configuration command implements a parameter at the global level which modifies the default cost calculation. The configured OSPF cost value can then be modified for a specific VLAN interface, particular subnet, or complete VLAN. The cost configured with the global command is then used as the default cost at VLAN level provided it is not explicitly configured locally. When the cost is explicitly configured at VLAN, this cost will take precedence over the global level cost.

- This global parameter, if configured, must be used as the default value:
  - when a new VLAN is associated with an OSPF area without cost configuration.
  - when an existing VLAN cost parameter is not configured.
  - when an existing VLAN cost parameter is unconfigured.

The global cost auto configuration is supported only for OSPFv2 and OSPFv3.

No existing functionality is modified except that the cost will be configured automatically from global command when not provided at VLAN level.

Supported Platforms:

3810 - KB
**router ospf reference-cost**

**Syntax**

```
router [ospf | ospfv3] reference-cost [0-16777215]
```

```
no router [ospf | ospfv3] reference-cost
```

**Description**

Configures the global level cost for OSPF at the VLAN level. The global cost will become the default cost.

The `no` form of the command removes the configured global cost for OSPF.

---

In the event that the cost is explicitly configured at the VLAN level, the locally configured cost will take precedence over the global level cost.

**Command context**

**Required context:** `config`

**Parameters**

- `ospf|ospfv3`  
  Select the correct version of OSPF.

- `0-16777215`  
  Specifies the global cost range for OSPF at the VLAN level.

**Use cases**

Use cases for the configuration of costs automatically from global level to VLAN level:

1. Verify that the globally configured cost is applied at the VLAN level when configured without explicit cost parameters.
2. Verify that the globally configured cost is applied on the VLAN level when the existing VLAN level cost is unconfigured.
3. Verify that the auto configured cost is considered in OSPF neighbors for cost-related OSPF operations (such as shortest path selection).
4. Verify that the existing command implementation is not affected due to an automatic cost configuration on both the global and the VLAN level.
5. Check that the configuration is applied properly after uploading and downloading a saved configuration through TFTP.
6. Verify the configuration by saving the configuration `write mem` and rebooting the system.
7. When the area is already configured for VLAN, only OSPF cost will be applied.
For general information about route policy, see Route policy overview on page 242.

**Using prefix lists**

Prefix lists are named lists of route prefixes. They are used to match routes for inclusion in or exclusion from route policies.

**Creating prefix list entries**

A prefix list can include one or more rules, each defined by a sequence number, permit or deny instruction, prefix, and range of allowed prefix lengths.

**Syntax:**

```
no [ip | ipv6 prefix-list name] [seq seq-num] [permit | deny prefix / prefix-length] [ge min-length] [le max-length]
```

Enters a route prefix into a prefix list.

- `[ip | ipv6]` Specifies a list of either IPv4 (IP) or IPv6 prefixes.
- `name` Specifies the name of the prefix list to which this prefix will be added. If the named list does not exist, this command creates it.
- `To add a prefix to an existing list, specify the name of that list.`
- `seq seq-num` Optionally specifies a sequence number for the entry.
- `permit` Permits the prefix when a successful match is made.
- `deny` Denies the prefix when a successful match is made.
- `prefix/prefix-length` Specifies an IPv4 or IPv6 network prefix and its mask length, in CIDR notation. For example: 10.1.4.1/24.
- `ge min-length` Specifies a minimum mask length of the prefix to match. `min-length` must have a value between 1 and 32 for IPv4, or a value between 1 and 128 for IPv6.
- `This value must be greater than or equal to prefix-length. If this optional parameter is not specified, its value defaults to prefix-length.`
- `le max-length` Specifies a maximum mask length of the prefix to match. `max-length` must have a value between 1 and 32 for IPv4, or a value between 1 and 128 for IPv6.
This value must be greater than or equal to min-length. If this optional parameter is not specified, its value defaults to prefix-length. (If you have specified a value for min-length that is greater than prefix-length, you must explicitly specify it with a max-length value that is greater than or equal to min-length.)

```
no [ip | ipv6 prefix-list name]
```

Deletes the entire prefix list identified by name.

```
no [ip | ipv6 prefix-list name] [seq seq-num]
```

Deletes the entry with the specified sequence number from the prefix list identified by name.

Individual prefix list entries are made using separate commands in the general configuration context. All entries that have the same prefix list name are part of the same prefix list. Thus, the following commands, taken from a show running-config listing, constitute two prefix lists.

```
  ip prefix-list "Odd"  seq 5 permit 10.1.1.1 255.255.255.0 ge 24 le 24
  ip prefix-list "Odd"  seq 10 deny 10.1.2.1 255.255.255.0 ge 24 le 24
  ip prefix-list "Odd"  seq 15 permit 10.1.3.1 255.255.255.0 ge 24 le 24
  ip prefix-list "Odd"  seq 20 deny 10.1.4.1 255.255.255.0 ge 24 le 24
  ip prefix-list "Even"  seq 5 permit 10.1.1.1 255.255.255.0 ge 24 le 24
  ip prefix-list "Even"  seq 10 permit 10.1.2.1 255.255.255.0 ge 24 le 24
  ip prefix-list "Even"  seq 15 deny 10.1.3.1 255.255.255.0 ge 24 le 24
  ip prefix-list "Even"  seq 20 permit 10.1.4.1 255.255.255.0 ge 24 le 24
```

Sequence numbers, which are optional, determine the order in which prefix list entries are evaluated during match operations. If you do not specify a sequence number for an entry, the switch uses a number that is 5 more than the highest sequence number already used in the list. (For the first entry in a prefix list, the default value of the sequence number is 5.) You can insert a new entry in a prefix list between two entries already in the list by specifying a sequence number for the new entry that is between the sequence numbers of the two existing entries.

**Entering a prefix list description**

Use the following command to enter a description string into an existing prefix list:

```
Syntax:
[ip | ipv6 prefix-list name] [seq seq-num description description-string]
```

Enters a description into a prefix list.

- **[ip | ipv6]**
  Specifies an IPv4 (IP) or IPv6 prefix list.

- **name**
  Specifies the name of the prefix list to which this description will be added. The prefix list must already exist.

- **seq seq-num**
  Optionally specifies a sequence number for the description entry. The description is attached to the prefix list entry identified by that sequence number. If the prefix list does not contain an entry with that sequence number, no description is entered.

  If you do not specify a sequence number, the description is attached to the first entry in the prefix list at the time the description is entered.

- **description-string**
  Specifies a description string of up to 80 characters.

  If you delete the entry to which the description is attached, the description is deleted also.
Viewing prefix lists

The `show ip prefix-list` command displays the content of prefix lists.

**Syntax:**

```
show [ip | ipv6 prefix-list] [name list-name] [summary | detail]
```

Displays the content of prefix lists.

- **[ip | ipv6]**
  Specifies an IPv4 (IP) or IPv6 prefix list.

- **name list-name**
  Specifies the name of the prefix list to display. If this parameter is omitted, all prefix lists are displayed.

- **[summary | detail]**
  If neither `summary` nor `detail` is specified, the listing displays the name of the prefix list and each entry in the list (not including descriptions.)

  If `summary` is specified, the listing displays the name of the list and a summary of the entries (but not the entries themselves.)

  If `detail` is specified, the listing displays the summary information, the description (if it exists), and the entries in the list.

**Example**

In a switch that contains two prefix lists, a standard display looks like this:

```
switch# show ip prefix-list

ip prefix-list Odd: 4 entries
   seq 5 permit 10.1.1.1/24 ge 24 le 24
   seq 10 deny 10.1.2.1/24 ge 24 le 24
   seq 15 permit 10.1.3.1/24 ge 24 le 24
   seq 20 deny 10.1.4.1/24 ge 24 le 24

ip prefix-list Even: 4 entries
   seq 5 deny 10.1.1.1/24 ge 24 le 24
   seq 10 permit 10.1.2.1/24 ge 24 le 24
   seq 15 deny 10.1.3.1/24 ge 24 le 24
   seq 20 permit 10.1.4.1/24 ge 24 le 24
```

A summary of the prefix lists looks like this:

```
switch# show ip prefix-list summary

ip prefix-list Odd: Count:4, Range-entries: 4,
   Sequences: 5 - 20

ip prefix-list Even: Count:4, Range-entries: 4,
   Sequences: 5 - 20
```

A detailed display of one of the prefix lists looks like this:

```
switch# show ip prefix-list name Even detail

ip prefix-list Even: Count:4, Range-entries: 4,
   Sequences: 5 - 20
   seq 5 deny 10.1.1.1/24 ge 24 le 24
      Description: Permit even-numbered subnets
   seq 10 permit 10.1.2.1/24 ge 24 le 24
```
Creating a route map

The route-map command creates a route map sequence. It specifies a route map name, a permit or deny instruction, and, optionally, a sequence number. All sequences that have the same route map name belong to the same route map. For more information, see Route maps on page 243.

Syntax:
route-map name [permit | deny] [seq seq-num]

Creates a route map and enters the route map context.

name
Specifies the name of the route map.

permit
Instructs the policy engine to permit the route if the match succeeds.

deny
Instructs the policy engine to deny the route if the match succeeds.

seq seq-num
Specifies a sequence number for the route-map. If a sequence number is not specified at the first instance of the route-map name command, the switch uses a default value of 10.

Deleting all or part of a route map

Use The no form of the route-map command to delete a sequence or an entire route map.

Syntax:
no route-map name [seq seq-num]

Deletes a route map or a route map sequence.

name
Specifies the name of the route map.

seq seq-num
Optional sequence number. Specifies a sequence to delete from the named route map.

If no sequence number is specified, the entire route map is deleted.

To delete a match or set clause from a route-map, first enter the context of that route map and then issue The no form of the clause to delete it.

Example

To delete the match metric 25 clause from sequence 20 of Map4, you would use the following commands:

```bash
switch(config)# route-map Map4 permit seq 20
switch(route-map-Map4-20)# no match metric 25
```

Viewing route maps

Syntax:
show route-map [name]

Displays the commands in all route maps or in a specified route map.

[name]
Optionally specifies the name of a route map to display. If no name is specified, all route maps are displayed. All sequences of a route map are displayed. For example:

```
switch(config)# show route-map Map3
  Routemap information

  route-map "Map3" permit seq 10
    match interface vlan 11 12 13
    match metric 25
    exit
  route-map "Map3" permit seq 20
    match interface vlan 21 22 23
    match metric 25
    exit
```

**Using match commands**

For more information, see [Match commands on page 244](#).

**Matching VLANs**

**Syntax:**

```
match interface vlan vid [vid]...
no match interface vlan vid [vid]...
```

Matches a VLAN interface. The `no` form of the command deletes the match clause from the sequence.

- `vid`
  Specifies the ID number of the VLAN to match.

- `[vid]...`
  Optional additional VLAN identifiers. A single command can specify multiple VLANs. A match succeeds if any of the VLANs matches (logical OR.)

**Matching prefix lists**

**Syntax:**

```
match [ip | ipv6] address prefix-list name
no match [ip | ipv6] address prefix-list name
```

Matches a prefix list. The `no` form of the command deletes the match clause from the sequence.

- `[ip | ipv6]`
  Specifies matching with a prefix list that contains either IPv4 (IP) or IPv6 addresses, respectively.

- `name`
  Specifies the name of the prefix list to match.

**Matching next-hop addresses**

**Syntax:**

```
no match [ip | ipv6 next-hop IP-addr | IPv6-addr] [IP-addr | IPv6-addr ...]
no match [ip | ipv6 next-hop prefix-list name]
```

Matches a next hop address. The `no` form of the command deletes the match clause from the sequence.

- `[ip | ipv6]`
  Specifies matching with either an IPv4 (IP) or IPv6 address, respectively.

- `[IP-addr | IPv6-addr]`
  Specifies the IPv4 (IP) or IPv6 address, respectively, to match with.
[IP-addr | IPv6-addr ...]
Optional additional addresses. A single command can specify multiple IPv4 (IP) or IPv6 addresses. A match succeeds if any of the addresses matches (logical OR.)

name
Specifies the name of a prefix list to match the next hop against.

Matching route sources
Syntax:
match [ip | ipv6] route-source prefix-list name
no match [ip | ipv6] route-source prefix-list name
Matches the address of an advertising router. The no form of the command deletes the match clause from the sequence.

[ip | ipv6]
Specifies matching with a prefix list that contains either IPv4 (IP) or IPv6 addresses, respectively.

name
Specifies the name of a prefix list to match the advertising router against.

Matching route metrics
Syntax:
match metric value
no match metric value
Matches the specified metric value with that of the route. The no form of the command deletes the match clause from the sequence.

value
Value of the route metric to match against. This is an integer value between 0 and the maximum number supported by the routing switch.

Matching metric types
Syntax:
match route-type external [type-1 | type-2]
no match route-type external [type-1 | type-2]
Matches an OSPF external route metric type. The no form of the command deletes the match clause from the sequence.

type-1
Matches against an OSPF external route with a type-1 metric.

type-2
Matches against an OSPF external route with a type-2 metric.

Matching source protocols
Syntax:
match source-protocol [connected | static | rip | ospf | ospfv3]
no match source-protocol [connected | static | rip | ospf | ospfv3]
Matches the protocol type of the destination prefix. The no form of the command deletes the match clause from the sequence.

connected
Matches directly connected routes.

static
Matches static routes.

rip
Matches RIP routes.

ospf
Matches OSPF routes.

ospfv3
Matches OSPFv3 routes.

Matching tags
Syntax:
match tag value
no match tag value
Matches the specified tag value with that of the route. The no form of the command deletes the match clause from the sequence.

value: Value of the route tag to match against. This is an integer value between 0 and the maximum number supported by the routing switch. The tag value is typically set by a set command on a different router.

Using set commands
The set commands described below are available for use in route maps. Multiple set commands may be used in a sequence of a route map.

Setting the next hop
Syntax:
set [ip | ipv6] next-hop [IP-addr | IPv6-addr]
no set [ip | ipv6] next-hop [IP-addr | IPv6-addr]
Sets a next hop address. The no form of the command deletes the set clause from the sequence.

[ip | ipv6]
Specifies setting either an IPv4 (IP) or IPv6 address, respectively.

[IP-addr | IPv6-addr]
Specifies the IPv4 (IP) or IPv6 address, respectively, to set.

Setting the route metric
Syntax:
set metric value
no set metric value
Sets the route metric to the specified value. The no form of the command deletes the set clause from the sequence.

value
Value to be set for the route metric. This is an integer value between 0 and the maximum number supported by the routing switch.

Setting the metric type
Syntax:
set metric-type external [type-1 | type-2]
no set metric-type external [type-1 | type-2]

Sets the metric type of an OSPF external route. The no form of the command deletes the set clause from the sequence.

type-1
Sets the metric type of an OSPF external route to type 1.

type-2
Sets the metric type of an OSPF external route to type 2.

Setting the tag value

Syntax:
set tag value
no set tag value

Sets the tag value of the route. The no form of the command deletes the set clause from the sequence.

value
Value of the route tag. This is an integer value between 0 and the maximum number supported by the routing switch.

Route policy overview

The route table in a routing switch contains routing paths to IP destinations. The traditional sources of the routing paths are:

- Directly connected destinations (no router hops)
- Static routes (manually configured by a network administrator)
- Routing protocols such as RIP or OSPF

Route policy provides an additional method for controlling entries in the route table. This approach applies predetermined policies to define how the routing switch accepts routes from peers, propagates routes to peers, and redistributes routes between different protocols. Route policy can often provide finer control and greater flexibility over route table entries than traditional methods.

Route policy is embodied in route maps, which are used to match destination routes according to IP addresses and other parameters. Optional set statements allow changing properties of the route depending on the match. Typical uses for route policy include filtering and redistribution of routes.
Configuring route policy

The steps in configuring a route policy are:

1. (Optional) Create any prefix lists you will use to select routes for your policy.
2. Create a route map.
3. Include match statements in your route map to define the selection criteria for routes.
4. (Optional) Include set statements in your route map to modify properties of your routes.
5. Apply the policy.

Route maps

Route maps are policy tools that are used to match destination prefixes, interfaces, or other route properties. Optionally, they may change the properties of the route, depending on the match.

The route map includes one or more sequences, each of which contains match statements and, optionally, set statements. When a route map is applied, its sequences are evaluated in order. If all the match statements in a sequence match the target route, the match succeeds and the route is permitted or denied according to the permit | deny instruction in the route-map command that defined the sequence; if the sequence contains set statements, they are applied to the target route. If any of the match statements in the sequence does not match the target route, the match fails and the next sequence in the route map is evaluated. If all the sequences fail to match the route, the route is denied.

If the named route map does not already exist, the route-map command creates the route map and enters the route map context. For example:

```
switch(config)# route-map Map1 permit
switch(route-map-Map1-10)#
```

At this point, you are ready to enter match and set commands, described below. When you have finished entering match and set commands, an exit command exits the route map context and returns to the general configuration context.
When entering **match** commands, most allow only one command of a given type in a sequence. (For instance, you can enter **match source-protocol rip** or **match source-protocol ospf**, but not both.) The exceptions are matching VLAN interfaces and next hops. Multiple

**match interface vlan vid**

commands are concatenated into a single command, and a match succeeds if any of the VLANs matches. For example, the following two route maps are equivalent:

```
switch(config)# route-map Map2 permit
switch(route-map=Map2-10)# match interface vlan 11
switch(route-map=Map2-10)# match interface vlan 12
switch(route-map=Map2-10)# match interface vlan 13
switch(route-map=Map2-10)# ex

switch(config)# route-map Map3 permit
switch(route-map=Map3-10)# match interface vlan 11 12 13
switch(route-map=Map3-10)# ex
```

Similarly, multiple instances of the **match ip next-hop IP-addr** and **match ipv6 next-hop IPv6-addr** commands are concatenated internally into single commands, respectively.

The general limitation of only one match command of a given type applies within a sequence. The same type of match command can be repeated in other sequences in the same route map.

All of the match clauses of the sequence must match for a match to succeed. (For this purpose, multiple **match interface vlan**, **match ip next-hop**, and **match ipv6 next-hop** clauses are treated as a single clause. In such a clause, the interfaces or next hops are treated in logical OR fashion: if there is a match with any one of them, the match clause succeeds.)

A match sequence that contains no match commands will permit all routes. (Such a sequence may be used in a route map that denies certain routes but permits all others.)

Like most **match** commands, **set** commands allow only one command of a given type in a sequence. So, for instance, if a match sequence is successful, you can set a metric of 23, but not metrics of 23 and 25 simultaneously.

To re-enter the context of an existing route map that has only one sequence (say, to add or delete match or set statements), the sequence number is optional: **route-map name permit | deny**. If the route-map has more than one sequence, the sequence number is required: **route-map name permit | deny seq seq-num**.

To create a new sequence in an existing route map (that is, under the same route map name), use the **route-map** command with a different sequence number. Sequence numbers are significant: they determine the order of evaluation of sequences in route maps—the sequence with the lowest number is evaluated first.

**Match commands**

The **match** commands described in this chapter are available for use in route maps.

Multiple **match** commands may be used in a sequence of a route map. For most commands, only one match of a given type is permitted in a sequence. For the **match interface vlan vid**, **match ip next-hop IP-addr**, and **match ipv6 next-hop IPv6-addr** commands, multiple instances of those commands are permitted in a single sequence, because all instances of those commands in a sequence are concatenated internally into single commands, respectively.

**Using route policy in route redistribution**
The following examples show some basic uses of route policy based on the figure below. (All subnets have 24-bit masks.)

**Figure 11 Network for redistribution example**

**Baseline: Intra-domain routing using default settings**

Each of the routing domains in Figure 11 is defined with simple VLANs and a basic routing configuration:

- In the RIP domains, the RIP protocol is assigned to each VLAN that a router connects to.
- Routers in the RIP domains redistribute connected routes—this is the default setting when RIP is enabled.
- For simplicity, all VLANs in the OSPF domain are assigned to the backbone area (area 0.)
- Border routers (North and South) implement both RIP and OSPF protocols.

The following listing shows the running configuration for the South router, the most complicated of the routers in this example. (Not only is the South router a border router, but it also has host computers connected directly to it in both RIP and OSPF domains.)
South(config)# show run

Running configuration:

; J8697A Configuration Editor; Created on release #K.15.01.0031

hostname "South"
module 1 type J8702A
module 3 type J9478A
ip routing
vlan 1
  name "DEFAULT_VLAN"
  untagged A19-A24,C13-C24
  ip address dhcp-bootp
  no untagged A1-A18,C1-C12
  exit
vlan 31
  name "VLAN31"
  untagged A1-A6
  ip address 10.3.31.2 255.255.255.0
  exit
vlan 33
  name "VLAN33"
  untagged A7-A12
  ip address 10.3.33.2 255.255.255.0
  exit
vlan 21
  name "VLAN21"
  untagged A13-A18
  ip address 10.2.21.1 255.255.255.0
  exit
vlan 37
  name "VLAN37"
  untagged C1-C6
  ip address 10.3.37.1 255.255.255.0
  exit
vlan 29
  name "VLAN29"
  untagged C7-C12
  ip address 10.2.29.1 255.255.255.0
  exit
router ospf
  area backbone
  exit
router rip
  redistribute connected
  exit
snmp-server community "public" unrestricted
vlan 21
  ip rip 10.2.21.1
  exit
vlan 29
  ip rip 10.2.29.1
  exit
vlan 31
  ip ospf 10.3.31.2 area backbone
  exit
vlan 33
  ip ospf 10.3.33.2 area backbone
  exit
vlan 37
  ip ospf 10.3.37.1 area backbone
  exit
Items of particular interest are:

- The `ip routing` command enables routing on the switch.
- The `router ospf` command enables OSPF routing on the switch. The `area backbone` command establishes the backbone area (area 0).
- The `router rip` command enables RIP routing on the switch. The `redistribute connected` command redistributes directly connected routes to all routers in the attached RIP domain.
- The `vlan` commands at the end of the configuration assign routing protocols to the VLANs. Additionally, they make area assignments for VLANs in the OSPF domain.

The other routers have analogous, if somewhat simpler, routing configurations. The Northwest, Northeast, and Southeast routers have only RIP enabled, and the East router has only OSPF enabled. The North router enables both routing protocols, but has fewer VLANs.

Listed below are the routing tables that result for three representative routers:

**South**
A border router attached to both RIP and OSPF domains.

**East**
A router within the OSPF domain.

**Southeast**
A router within the RIP domain.

```
South(config)# show ip route

<table>
<thead>
<tr>
<th>Destination</th>
<th>Gateway</th>
<th>VLAN</th>
<th>Type</th>
<th>Sub-Type</th>
<th>Metric</th>
<th>Dist.</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.2.21.0/24</td>
<td>VLAN21</td>
<td>21</td>
<td>connected</td>
<td>1</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>10.2.22.0/24</td>
<td>10.2.21.2</td>
<td>21</td>
<td>rip</td>
<td></td>
<td>2</td>
<td>120</td>
</tr>
<tr>
<td>10.2.23.0/24</td>
<td>10.2.21.2</td>
<td>21</td>
<td>rip</td>
<td></td>
<td>2</td>
<td>120</td>
</tr>
<tr>
<td>10.2.29.0/24</td>
<td>VLAN29</td>
<td>29</td>
<td>connected</td>
<td>1</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>10.3.31.0/24</td>
<td>VLAN31</td>
<td>31</td>
<td>connected</td>
<td>1</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>10.3.32.0/24</td>
<td>10.3.31.1</td>
<td>31</td>
<td>ospf</td>
<td>IntraArea</td>
<td>2</td>
<td>110</td>
</tr>
<tr>
<td>10.3.32.0/24</td>
<td>10.3.31.1</td>
<td>33</td>
<td>ospf</td>
<td>IntraArea</td>
<td>2</td>
<td>110</td>
</tr>
<tr>
<td>10.3.33.0/24</td>
<td>VLAN33</td>
<td>33</td>
<td>connected</td>
<td>1</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>10.3.34.0/24</td>
<td>10.3.33.1</td>
<td>33</td>
<td>ospf</td>
<td>IntraArea</td>
<td>2</td>
<td>110</td>
</tr>
<tr>
<td>10.3.37.0/24</td>
<td>VLAN37</td>
<td>37</td>
<td>connected</td>
<td>1</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>127.0.0.0/8</td>
<td>reject</td>
<td>static</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>127.0.0.1/32</td>
<td>lo0</td>
<td>connected</td>
<td>1</td>
<td>0</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

East(config)# show ip route

<table>
<thead>
<tr>
<th>Destination</th>
<th>Gateway</th>
<th>VLAN</th>
<th>Type</th>
<th>Sub-Type</th>
<th>Metric</th>
<th>Dist.</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.3.31.0/24</td>
<td>10.3.32.1</td>
<td>32</td>
<td>ospf</td>
<td>IntraArea</td>
<td>2</td>
<td>110</td>
</tr>
<tr>
<td>10.3.31.0/24</td>
<td>10.3.32.2</td>
<td>32</td>
<td>ospf</td>
<td>IntraArea</td>
<td>2</td>
<td>110</td>
</tr>
<tr>
<td>10.3.32.0/24</td>
<td>VLAN32</td>
<td>32</td>
<td>connected</td>
<td>1</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>10.3.33.0/24</td>
<td>VLAN33</td>
<td>33</td>
<td>connected</td>
<td>1</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>10.3.34.0/24</td>
<td>VLAN34</td>
<td>34</td>
<td>connected</td>
<td>1</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>10.3.37.0/24</td>
<td>10.3.33.2</td>
<td>33</td>
<td>ospf</td>
<td>IntraArea</td>
<td>2</td>
<td>110</td>
</tr>
<tr>
<td>127.0.0.0/8</td>
<td>reject</td>
<td>static</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>127.0.0.1/32</td>
<td>lo0</td>
<td>connected</td>
<td>1</td>
<td>0</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
```
Southeast(config)# show ip route

<table>
<thead>
<tr>
<th>Destination</th>
<th>Gateway</th>
<th>VLAN</th>
<th>Type</th>
<th>Sub-Type</th>
<th>Metric</th>
<th>Dist.</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.2.21.0/24</td>
<td>VLAN21</td>
<td>21</td>
<td>connected</td>
<td>1</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>10.2.22.0/24</td>
<td>VLAN22</td>
<td>22</td>
<td>connected</td>
<td>1</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>10.2.23.0/24</td>
<td>VLAN23</td>
<td>23</td>
<td>connected</td>
<td>1</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>10.2.29.0/24</td>
<td>10.2.21.1</td>
<td>21</td>
<td>rip</td>
<td>2</td>
<td>120</td>
<td></td>
</tr>
<tr>
<td>10.3.31.0/24</td>
<td>10.2.21.1</td>
<td>21</td>
<td>rip</td>
<td>2</td>
<td>120</td>
<td></td>
</tr>
<tr>
<td>10.3.33.0/24</td>
<td>10.2.21.1</td>
<td>21</td>
<td>rip</td>
<td>2</td>
<td>120</td>
<td></td>
</tr>
<tr>
<td>10.3.37.0/24</td>
<td>10.2.21.1</td>
<td>21</td>
<td>rip</td>
<td>2</td>
<td>120</td>
<td></td>
</tr>
<tr>
<td>127.0.0.0/8</td>
<td>reject</td>
<td>static</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>127.0.0.1/32</td>
<td>lo0</td>
<td>connected</td>
<td>1</td>
<td>0</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

With this configuration, the routers and host computers in each routing domain are able to communicate with all other routers and hosts in that domain. In addition, the routers and hosts in the RIP domains can communicate with all interfaces of the adjacent border router and with hosts attached to those interfaces. (To prevent that cross-domain communication, you would remove the `redistribute connected` command from the router rip context.) Beyond those connected routes on the RIP side, there is no inter-domain communication. Thus, host Z can ping host X and host L, but not host M or host B. And host M can ping host L, but not host X or host Y or host A. And so on.

**Basic inter-domain protocol redistribution**

Route redistribution allows border routers to distribute routes between adjacent routing domains. Thus, the North router can redistribute routes from the northern RIP domain to the OSPF domain and from the OSPF domain to the northern RIP domain. Similarly, the South router can redistribute routes from the southern RIP domain to the OSPF domain and from the OSPF domain to the southern RIP domain. And if both the North and South routers have redistribution enabled in both directions at the same time, the routes that are redistributed from the RIP domains to the OSPF domain will be further distributed to the opposite RIP domain, and routers and hosts in all domains will be able to communicate with each other. (Some subtle complications are explained below.)

For example, in the North and South routers you might add a `redistribute rip` command to the router ospf context and a `redistribute ospf` command to the router rip context, like this:

```
  .
  .
  router ospf
  area backbone
  redistribute rip
  exit
  router rip
  redistribute connected
  redistribute ospf
  exit
  .
  .
```

This causes extensive redistribution of routes within all three routing domains, adding a large number of routes to the route tables of all the routers. For example, the route table in the East router adds routes to subnets in both RIP domains, and looks like this:
East(config)# show ip route

<table>
<thead>
<tr>
<th>Destination</th>
<th>Gateway</th>
<th>VLAN</th>
<th>Type</th>
<th>Sub-Type</th>
<th>Metric</th>
<th>Dist.</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.1.11.0/24</td>
<td>10.3.32.1</td>
<td>32</td>
<td>ospf</td>
<td>External2</td>
<td>10</td>
<td>110</td>
</tr>
<tr>
<td>10.1.12.0/24</td>
<td>10.3.32.1</td>
<td>32</td>
<td>ospf</td>
<td>External2</td>
<td>10</td>
<td>110</td>
</tr>
<tr>
<td>10.1.13.0/24</td>
<td>10.3.32.1</td>
<td>32</td>
<td>ospf</td>
<td>External2</td>
<td>10</td>
<td>110</td>
</tr>
<tr>
<td>10.1.14.0/24</td>
<td>10.3.32.1</td>
<td>32</td>
<td>ospf</td>
<td>External2</td>
<td>10</td>
<td>110</td>
</tr>
<tr>
<td>10.2.22.0/24</td>
<td>10.3.33.2</td>
<td>33</td>
<td>ospf</td>
<td>External2</td>
<td>10</td>
<td>110</td>
</tr>
<tr>
<td>10.2.23.0/24</td>
<td>10.3.33.2</td>
<td>33</td>
<td>ospf</td>
<td>External2</td>
<td>10</td>
<td>110</td>
</tr>
<tr>
<td>10.3.31.0/24</td>
<td>10.3.32.1</td>
<td>32</td>
<td>ospf</td>
<td>IntraArea</td>
<td>2</td>
<td>110</td>
</tr>
<tr>
<td>10.3.31.0/24</td>
<td>10.3.32.1</td>
<td>32</td>
<td>ospf</td>
<td>IntraArea</td>
<td>2</td>
<td>110</td>
</tr>
<tr>
<td>10.3.32.0/24</td>
<td>32</td>
<td></td>
<td>connected</td>
<td>1</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>10.3.33.0/24</td>
<td>33</td>
<td></td>
<td>connected</td>
<td>1</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>10.3.34.0/24</td>
<td>34</td>
<td></td>
<td>connected</td>
<td>1</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>10.3.37.0/24</td>
<td>33</td>
<td></td>
<td>ospf</td>
<td>IntraArea</td>
<td>2</td>
<td>110</td>
</tr>
<tr>
<td>127.0.0.0/8</td>
<td>reject</td>
<td>static</td>
<td></td>
<td></td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>127.0.0.1/32</td>
<td>lo0</td>
<td></td>
<td>connected</td>
<td>1</td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>

But this route table does not include all the possible routes in all domains: routes to subnets 10.1.15.x, 10.1.16.x, 10.2.21.x, and 10.2.29.x (VLANs 15, 16, 21, and 29) are missing. Host computer M cannot ping host X because there is no route to it, though it can ping through the "invisible" South router to host Y or host Z.

The problem is that those missing subnets are directly connected to the North and South border routers, and directly connected routes must be explicitly redistributed with a redistribute connected command even though they are RIP routes and RIP routes were redistributed. So by adding

```
redistribute connected
```

commands to the router ospf contexts of the North and South routers, like this:

```
. .
router ospf
  area backbone
  redistribute connected
  redistribute rip
  exit
. .
```

All existing routes are redistributed and the route table for the East router is now complete:

East(config)# show ip route

<table>
<thead>
<tr>
<th>Destination</th>
<th>Gateway</th>
<th>VLAN</th>
<th>Type</th>
<th>Sub-Type</th>
<th>Metric</th>
<th>Dist.</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.1.11.0/24</td>
<td>10.3.32.1</td>
<td>32</td>
<td>ospf</td>
<td>External2</td>
<td>10</td>
<td>110</td>
</tr>
<tr>
<td>10.1.12.0/24</td>
<td>10.3.32.1</td>
<td>32</td>
<td>ospf</td>
<td>External2</td>
<td>10</td>
<td>110</td>
</tr>
<tr>
<td>10.1.13.0/24</td>
<td>10.3.32.1</td>
<td>32</td>
<td>ospf</td>
<td>External2</td>
<td>10</td>
<td>110</td>
</tr>
<tr>
<td>10.1.14.0/24</td>
<td>10.3.32.1</td>
<td>32</td>
<td>ospf</td>
<td>External2</td>
<td>10</td>
<td>110</td>
</tr>
<tr>
<td>10.1.15.0/24</td>
<td>10.3.32.1</td>
<td>32</td>
<td>ospf</td>
<td>External2</td>
<td>10</td>
<td>110</td>
</tr>
<tr>
<td>10.1.16.0/24</td>
<td>10.3.32.1</td>
<td>32</td>
<td>ospf</td>
<td>External2</td>
<td>10</td>
<td>110</td>
</tr>
<tr>
<td>10.2.21.0/24</td>
<td>10.3.33.2</td>
<td>33</td>
<td>ospf</td>
<td>External2</td>
<td>10</td>
<td>110</td>
</tr>
<tr>
<td>10.2.22.0/24</td>
<td>10.3.33.2</td>
<td>33</td>
<td>ospf</td>
<td>External2</td>
<td>10</td>
<td>110</td>
</tr>
<tr>
<td>10.2.23.0/24</td>
<td>10.3.33.2</td>
<td>33</td>
<td>ospf</td>
<td>External2</td>
<td>10</td>
<td>110</td>
</tr>
<tr>
<td>10.2.29.0/24</td>
<td>10.3.33.2</td>
<td>33</td>
<td>ospf</td>
<td>External2</td>
<td>10</td>
<td>110</td>
</tr>
<tr>
<td>10.3.31.0/24</td>
<td>10.3.32.1</td>
<td>32</td>
<td>ospf</td>
<td>IntraArea</td>
<td>2</td>
<td>110</td>
</tr>
<tr>
<td>10.3.31.0/24</td>
<td>10.3.32.1</td>
<td>32</td>
<td>ospf</td>
<td>IntraArea</td>
<td>2</td>
<td>110</td>
</tr>
</tbody>
</table>
Host L can now ping host X and, indeed, any other host in any of the three routing domains.

**Finer control of inter-domain routing using route policy**

The wide variety of match types available with route policy allows you to make finer distinctions when distributing routes across routing domain boundaries.

Suppose that you want to limit the distribution of the "non-connected" routes in the northern RIP domain to the "odd-numbered" prefixes—that is, to 10.1.11.x and 10.1.13.x. You can accomplish that by creating a prefix list:

```
ip prefix-list "Odds" seq 5 permit 10.1.11.1 255.255.255.0 ge 24 le 24
ip prefix-list "Odds" seq 10 permit 10.1.13.1 255.255.255.0 ge 24 le 24
```

Then matching that prefix-list in a route map:

```
route-map "PermitOdds" permit seq 10
  match ip address prefix-list "Odds"
  exit
```

And finally applying that route map to the redistribution of RIP routes in the North router:

```
routerr ospf
  area backbone
  redistribute connected
  redistribute rip route-map "PermitOdds"
  exit
```

The result of this is to permit redistribution of routes 10.1.11.x and 10.1.13.x, and to deny redistribution of routes 10.1.12.x and 10.1.14.x. (Routes 10.1.15.x and 10.1.16.x are redistributed by the redistribute connected command.) This occurs throughout the OSPF domain, and is propagated through redistribution by the South router into the southern RIP domain.

For instance, in the OSPF domain the route map of the East router becomes:

```
East(config)# show ip route

<table>
<thead>
<tr>
<th>Destination</th>
<th>Gateway</th>
<th>VLAN Type</th>
<th>Sub-Type</th>
<th>Metric</th>
<th>Dist.</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.1.11.0/24</td>
<td>10.3.32.1</td>
<td>32</td>
<td>ospf</td>
<td>10</td>
<td>110</td>
</tr>
<tr>
<td>10.1.13.0/24</td>
<td>10.3.32.1</td>
<td>32</td>
<td>ospf</td>
<td>10</td>
<td>110</td>
</tr>
<tr>
<td>10.1.15.0/24</td>
<td>10.3.32.1</td>
<td>32</td>
<td>ospf</td>
<td>10</td>
<td>110</td>
</tr>
<tr>
<td>10.1.16.0/24</td>
<td>10.3.32.1</td>
<td>32</td>
<td>ospf</td>
<td>10</td>
<td>110</td>
</tr>
<tr>
<td>10.2.21.0/24</td>
<td>10.3.33.2</td>
<td>33</td>
<td>ospf</td>
<td>10</td>
<td>110</td>
</tr>
<tr>
<td>10.2.22.0/24</td>
<td>10.3.33.2</td>
<td>33</td>
<td>ospf</td>
<td>10</td>
<td>110</td>
</tr>
<tr>
<td>10.2.23.0/24</td>
<td>10.3.33.2</td>
<td>33</td>
<td>ospf</td>
<td>10</td>
<td>110</td>
</tr>
<tr>
<td>10.2.29.0/24</td>
<td>10.3.33.2</td>
<td>33</td>
<td>ospf</td>
<td>10</td>
<td>110</td>
</tr>
<tr>
<td>10.3.31.0/24</td>
<td>10.3.32.1</td>
<td>32</td>
<td>IntraArea</td>
<td>2</td>
<td>110</td>
</tr>
<tr>
<td>10.3.31.0/24</td>
<td>10.3.33.2</td>
<td>33</td>
<td>IntraArea</td>
<td>2</td>
<td>110</td>
</tr>
<tr>
<td>10.3.31.0/24</td>
<td>VLAN32</td>
<td>32</td>
<td>connected</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>10.3.33.0/24</td>
<td>VLAN33</td>
<td>33</td>
<td>connected</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>10.3.34.0/24</td>
<td>VLAN34</td>
<td>34</td>
<td>connected</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>10.3.37.0/24</td>
<td>10.3.33.2</td>
<td>33</td>
<td>IntraArea</td>
<td>2</td>
<td>110</td>
</tr>
</tbody>
</table>
```
In the southern RIP domain, the route map of the Southeast router becomes:

Southeast(config)# show ip route

<table>
<thead>
<tr>
<th>Destination</th>
<th>Gateway</th>
<th>VLAN</th>
<th>Type</th>
<th>Sub-Type</th>
<th>Metric</th>
<th>Dist.</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.1.11.0/24</td>
<td>10.2.21.1</td>
<td>21</td>
<td>rip</td>
<td></td>
<td>2</td>
<td>120</td>
</tr>
<tr>
<td>10.1.13.0/24</td>
<td>10.2.21.1</td>
<td>21</td>
<td>rip</td>
<td></td>
<td>2</td>
<td>120</td>
</tr>
<tr>
<td>10.1.15.0/24</td>
<td>10.2.21.1</td>
<td>21</td>
<td>rip</td>
<td></td>
<td>2</td>
<td>120</td>
</tr>
<tr>
<td>10.1.16.0/24</td>
<td>10.2.21.1</td>
<td>21</td>
<td>rip</td>
<td></td>
<td>2</td>
<td>120</td>
</tr>
<tr>
<td>10.2.21.0/24</td>
<td>VLAN21</td>
<td>21</td>
<td>connected</td>
<td>1</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>10.2.22.0/24</td>
<td>VLAN22</td>
<td>22</td>
<td>connected</td>
<td>1</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>10.2.23.0/24</td>
<td>VLAN23</td>
<td>23</td>
<td>connected</td>
<td>1</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>10.2.29.0/24</td>
<td>10.2.21.1</td>
<td>21</td>
<td>rip</td>
<td></td>
<td>2</td>
<td>120</td>
</tr>
<tr>
<td>10.3.31.0/24</td>
<td>10.2.21.1</td>
<td>21</td>
<td>rip</td>
<td></td>
<td>2</td>
<td>120</td>
</tr>
<tr>
<td>10.3.32.0/24</td>
<td>10.2.21.1</td>
<td>21</td>
<td>rip</td>
<td></td>
<td>2</td>
<td>120</td>
</tr>
<tr>
<td>10.3.33.0/24</td>
<td>10.2.21.1</td>
<td>21</td>
<td>rip</td>
<td></td>
<td>2</td>
<td>120</td>
</tr>
<tr>
<td>10.3.34.0/24</td>
<td>10.2.21.1</td>
<td>21</td>
<td>rip</td>
<td></td>
<td>2</td>
<td>120</td>
</tr>
<tr>
<td>10.3.37.0/24</td>
<td>10.2.21.1</td>
<td>21</td>
<td>rip</td>
<td></td>
<td>2</td>
<td>120</td>
</tr>
<tr>
<td>127.0.0.0/8</td>
<td>reject</td>
<td>static</td>
<td></td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>127.0.0.1/32</td>
<td>lo0</td>
<td>connected</td>
<td></td>
<td>1</td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>

To not lose the "even-numbered" routes (10.1.12.x and 10.1.14.x) in the OSPF domain, reinstate the original redistribution in the North router:

```
router ospf
  area backbone
  redistribute connected
  redistribute rip
  exit
```

And move the prefix list, route map, and redistribution from the North router to the South router. To get the same distribution of routes from the northern RIP to the southern RIP domain, add the 10.1.15.x and 10.1.16.x routes to the prefix list—they will not be redistributed by the `redistribute connected` command because they are not directly connected to the South router. The prefix list would expand to:

```
ip prefix-list "Odds" seq 5 permit 10.1.11.1 255.255.255.0 ge 24 le 24
ip prefix-list "Odds" seq 10 permit 10.1.13.1 255.255.255.0 ge 24 le 24
ip prefix-list "Odds" seq 15 permit 10.1.15.1 255.255.255.0 ge 24 le 24
ip prefix-list "Odds" seq 20 permit 10.1.16.1 255.255.255.0 ge 24 le 24
```

The route map would move from North to South with no changes:

```
route-map "Odds" permit seq 10
  match ip address prefix-list "PermitOdds"
  exit
```

And the route redistribution would move from the `router ospf` context to the `router rip` context:

```
router rip
  redistribute connected
  redistribute ospf route-map "PermitOdds"
  exit
```

This has the desired effect of redistributing all the routes in the OSPF domain, as indicated by the East router's route table:
East(config)# show ip route

IP Route Entries

<table>
<thead>
<tr>
<th>Destination</th>
<th>Gateway</th>
<th>VLAN</th>
<th>Type</th>
<th>Sub-Type</th>
<th>Metric</th>
<th>Dist.</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.1.11.0/24</td>
<td>10.3.32.1</td>
<td>32</td>
<td>ospf</td>
<td>External2</td>
<td>10</td>
<td>110</td>
</tr>
<tr>
<td>10.1.12.0/24</td>
<td>10.3.32.1</td>
<td>32</td>
<td>ospf</td>
<td>External2</td>
<td>10</td>
<td>110</td>
</tr>
<tr>
<td>10.1.13.0/24</td>
<td>10.3.32.1</td>
<td>32</td>
<td>ospf</td>
<td>External2</td>
<td>10</td>
<td>110</td>
</tr>
<tr>
<td>10.1.14.0/24</td>
<td>10.3.32.1</td>
<td>32</td>
<td>ospf</td>
<td>External2</td>
<td>10</td>
<td>110</td>
</tr>
<tr>
<td>10.1.15.0/24</td>
<td>10.3.32.1</td>
<td>32</td>
<td>ospf</td>
<td>External2</td>
<td>10</td>
<td>110</td>
</tr>
<tr>
<td>10.1.16.0/24</td>
<td>10.3.32.1</td>
<td>32</td>
<td>ospf</td>
<td>External2</td>
<td>10</td>
<td>110</td>
</tr>
<tr>
<td>10.1.21.0/24</td>
<td>10.3.33.2</td>
<td>33</td>
<td>ospf</td>
<td>External2</td>
<td>10</td>
<td>110</td>
</tr>
<tr>
<td>10.1.22.0/24</td>
<td>10.3.33.2</td>
<td>33</td>
<td>ospf</td>
<td>External2</td>
<td>10</td>
<td>110</td>
</tr>
<tr>
<td>10.1.23.0/24</td>
<td>10.3.33.2</td>
<td>33</td>
<td>ospf</td>
<td>External2</td>
<td>10</td>
<td>110</td>
</tr>
<tr>
<td>10.3.31.0/24</td>
<td>10.3.32.1</td>
<td>32</td>
<td>ospf</td>
<td>IntraArea</td>
<td>2</td>
<td>110</td>
</tr>
<tr>
<td>10.3.31.0/24</td>
<td>10.3.33.2</td>
<td>33</td>
<td>ospf</td>
<td>IntraArea</td>
<td>2</td>
<td>110</td>
</tr>
<tr>
<td>10.3.32.0/24</td>
<td>VLAN32</td>
<td>32</td>
<td>connected</td>
<td></td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>10.3.33.0/24</td>
<td>VLAN33</td>
<td>33</td>
<td>connected</td>
<td></td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>10.3.34.0/24</td>
<td>VLAN34</td>
<td>34</td>
<td>connected</td>
<td></td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>10.3.37.0/24</td>
<td>10.3.33.2</td>
<td>33</td>
<td>ospf</td>
<td>IntraArea</td>
<td>2</td>
<td>110</td>
</tr>
<tr>
<td>127.0.0.0/8</td>
<td>reject</td>
<td>static</td>
<td></td>
<td></td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>127.0.0.1/32</td>
<td>lo0</td>
<td>connected</td>
<td></td>
<td></td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

However, it falls short in the southern RIP domain. The northern RIP routes are distributed as expected, but some of the routes from the OSPF domain are missing—10.3.32.x and 10.3.34.x. Here is the Southeast router's route table:

Southeast(config)# show ip route

IP Route Entries

<table>
<thead>
<tr>
<th>Destination</th>
<th>Gateway</th>
<th>VLAN</th>
<th>Type</th>
<th>Sub-Type</th>
<th>Metric</th>
<th>Dist.</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.1.11.0/24</td>
<td>10.2.21.1</td>
<td>21</td>
<td>rip</td>
<td></td>
<td>2</td>
<td>120</td>
</tr>
<tr>
<td>10.1.12.0/24</td>
<td>10.2.21.1</td>
<td>21</td>
<td>rip</td>
<td></td>
<td>2</td>
<td>120</td>
</tr>
<tr>
<td>10.1.13.0/24</td>
<td>10.2.21.1</td>
<td>21</td>
<td>rip</td>
<td></td>
<td>2</td>
<td>120</td>
</tr>
<tr>
<td>10.1.15.0/24</td>
<td>10.2.21.1</td>
<td>21</td>
<td>rip</td>
<td></td>
<td>2</td>
<td>120</td>
</tr>
<tr>
<td>10.1.16.0/24</td>
<td>VLAN21</td>
<td>21</td>
<td>connected</td>
<td></td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>10.2.21.0/24</td>
<td>VLAN22</td>
<td>22</td>
<td>connected</td>
<td></td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>10.2.22.0/24</td>
<td>VLAN23</td>
<td>23</td>
<td>connected</td>
<td></td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>10.2.23.0/24</td>
<td>10.2.21.1</td>
<td>21</td>
<td>rip</td>
<td></td>
<td>2</td>
<td>120</td>
</tr>
<tr>
<td>10.3.31.0/24</td>
<td>10.2.21.1</td>
<td>21</td>
<td>rip</td>
<td></td>
<td>2</td>
<td>120</td>
</tr>
<tr>
<td>10.3.32.0/24</td>
<td>10.2.21.1</td>
<td>21</td>
<td>rip</td>
<td></td>
<td>2</td>
<td>120</td>
</tr>
<tr>
<td>10.3.37.0/24</td>
<td>10.2.21.1</td>
<td>21</td>
<td>rip</td>
<td></td>
<td>2</td>
<td>120</td>
</tr>
<tr>
<td>127.0.0.0/8</td>
<td>reject</td>
<td>static</td>
<td></td>
<td></td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>127.0.0.1/32</td>
<td>lo0</td>
<td>connected</td>
<td></td>
<td></td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

You can solve this problem by adding a second sequence to the route map to deal with the routes from the OSPF domain. The expanded route map becomes:

route-map "PermitOdds" permit seq 10
match ip address prefix-list "Odds"
exit
route-map "PermitOdds" permit seq 20
match source-protocol ospf
exit

Now all the desired routes show up in the Southeast router's route table:

Southeast(config)# show ip route

IP Route Entries
In addition to using route maps to filter routes, you can also use them to apply properties to the routes. For example, to apply a route metric when redistributing routes from the northern RIP domain to the OSPF domain, you could apply the metric with a `set metric` command in a route map in the North router:

```
route-map "Metric25" permit seq 10
  match source-protocol rip
  set metric 25
exit
```

Then you could redistribute from the `router ospf` context:

```
router ospf
  area backbone
  redistribute connected
  redistribute rip route-map "Metric25"
exit
```

The results are displayed in the Metric column of the East router’s route map:

```
East(config)# show ip route

<table>
<thead>
<tr>
<th>Destination</th>
<th>Gateway</th>
<th>VLAN</th>
<th>Type</th>
<th>Sub-Type</th>
<th>Metric</th>
<th>Dist.</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.1.11.0/24</td>
<td>10.3.32.1</td>
<td>32</td>
<td>ospf</td>
<td>External2</td>
<td>25</td>
<td>110</td>
</tr>
<tr>
<td>10.1.12.0/24</td>
<td>10.3.32.1</td>
<td>32</td>
<td>ospf</td>
<td>External2</td>
<td>25</td>
<td>110</td>
</tr>
<tr>
<td>10.1.13.0/24</td>
<td>10.3.32.1</td>
<td>32</td>
<td>ospf</td>
<td>External2</td>
<td>25</td>
<td>110</td>
</tr>
<tr>
<td>10.1.14.0/24</td>
<td>10.3.32.1</td>
<td>32</td>
<td>ospf</td>
<td>External2</td>
<td>25</td>
<td>110</td>
</tr>
<tr>
<td>10.1.15.0/24</td>
<td>10.3.32.1</td>
<td>32</td>
<td>ospf</td>
<td>External2</td>
<td>25</td>
<td>110</td>
</tr>
<tr>
<td>10.1.16.0/24</td>
<td>10.3.32.1</td>
<td>32</td>
<td>ospf</td>
<td>External2</td>
<td>25</td>
<td>110</td>
</tr>
<tr>
<td>10.2.21.0/24</td>
<td>10.3.33.2</td>
<td>33</td>
<td>ospf</td>
<td>External2</td>
<td>10</td>
<td>110</td>
</tr>
<tr>
<td>10.2.22.0/24</td>
<td>10.3.33.2</td>
<td>33</td>
<td>ospf</td>
<td>External2</td>
<td>10</td>
<td>110</td>
</tr>
<tr>
<td>10.2.23.0/24</td>
<td>10.3.33.2</td>
<td>33</td>
<td>ospf</td>
<td>External2</td>
<td>10</td>
<td>110</td>
</tr>
<tr>
<td>10.3.31.0/24</td>
<td>10.3.32.1</td>
<td>32</td>
<td>ospf</td>
<td>IntraArea</td>
<td>2</td>
<td>110</td>
</tr>
<tr>
<td>10.3.32.0/24</td>
<td>10.3.33.2</td>
<td>33</td>
<td>ospf</td>
<td>IntraArea</td>
<td>2</td>
<td>110</td>
</tr>
<tr>
<td>10.3.33.0/24</td>
<td>10.3.32.1</td>
<td>32</td>
<td>ospf</td>
<td>IntraArea</td>
<td>2</td>
<td>110</td>
</tr>
<tr>
<td>10.3.34.0/24</td>
<td>10.3.32.1</td>
<td>32</td>
<td>ospf</td>
<td>IntraArea</td>
<td>2</td>
<td>110</td>
</tr>
<tr>
<td>127.0.0.0/8</td>
<td>reject</td>
<td></td>
<td>static</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>127.0.0.1/32</td>
<td>lo0</td>
<td></td>
<td>connected</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
```
Redistribution using tags

Tags provide an alternative method for redistributing routes. For instance, you can set tags when redistributing routes into a domain and then use those tags for matches when redistributing those routes out of the domain. In the following example, tags are set as the routes pass through the North router from the northern RIP domain to the OSPF domain, and those tags are used for matching when the routes pass out of the OSPF domain through the South router to the southern RIP domain.

Establish prefix lists on the North router to separate the "odd" and "even" routes:

```plaintext
ip prefix-list "Odds" seq 5 permit 10.1.11.1 255.255.255.0 ge 24 le 24
ip prefix-list "Odds" seq 10 permit 10.1.13.1 255.255.255.0 ge 24 le 24
ip prefix-list "Evens" seq 5 permit 10.1.12.1 255.255.255.0 ge 24 le 24
ip prefix-list "Evens" seq 10 permit 10.1.14.1 255.255.255.0 ge 24 le 24
```

Then set up a route map with separate sequences to tag the odd and even routes:

```plaintext
route-map "TagIn" permit seq 10
  match ip address prefix-list "Odds"
  set tag 1
  exit
route-map "TagIn" permit seq 20
  match ip address prefix-list "Evens"
  set tag 2
  exit
```

Set up a separate route map to match the connected routes, and assign the same tag value you used for the odd routes. This allows you to propagate both the odd and the connected routes, but not the even routes, to the southern RIP domain.

```plaintext
route-map "TagConn" permit seq 10
  match source-protocol connected
  set tag 1
  exit
```

Redistribute the routes to the OSPF domain using the route maps:

```plaintext
router ospf
  area backbone
  redistribute connected route-map "TagConn"
  redistribute rip route-map "TagIn"
  exit
```

On the South router set up a route map with three sequences:

- One to permit routes with tag values of 1
- One to deny routes with tag values of 2
- One to permit OSPF routes (this propagates all the routes from the OSPF domain)

The route map looks like this:

```plaintext
route-map "TagOut" permit seq 10
  match tag 1
  exit
route-map "TagOut" deny seq 20
  match tag 2
  exit
route-map "TagOut" permit seq 30
  match source-protocol ospf
  exit
```
This arrangement permits the odd routes from the northern RIP domain and the RIP routes that were connected to the North router. It denies the even routes from the northern RIP domain, and it permits the OSPF routes. The route table from the Southeast router shows the results:

Southeast(config)# show ip route

<table>
<thead>
<tr>
<th>IP Route Entries</th>
</tr>
</thead>
<tbody>
<tr>
<td>Destination</td>
</tr>
<tr>
<td>10.1.11.0/24</td>
</tr>
<tr>
<td>10.1.13.0/24</td>
</tr>
<tr>
<td>10.1.15.0/24</td>
</tr>
<tr>
<td>10.1.16.0/24</td>
</tr>
<tr>
<td>10.2.21.0/24</td>
</tr>
<tr>
<td>10.2.22.0/24</td>
</tr>
<tr>
<td>10.2.23.0/24</td>
</tr>
<tr>
<td>10.2.29.0/24</td>
</tr>
<tr>
<td>10.3.31.0/24</td>
</tr>
<tr>
<td>10.3.32.0/24</td>
</tr>
<tr>
<td>10.3.33.0/24</td>
</tr>
<tr>
<td>10.3.34.0/24</td>
</tr>
<tr>
<td>10.3.37.0/24</td>
</tr>
<tr>
<td>127.0.0.0/8</td>
</tr>
<tr>
<td>127.0.0.1/32</td>
</tr>
</tbody>
</table>
The ICMP Router Discovery Protocol (IRDP) is used by routing switches to advertise the IP addresses of their router interfaces to directly attached hosts. IRDP is disabled by default. You can enable the feature on a global basis or on an individual VLAN interface basis.

**Configuring IRDP**

When IRDP is enabled, the routing switch periodically sends Router Advertisement messages out the IP interfaces on which the feature is enabled. The messages advertise the routing switch’s IP addresses to directly attached hosts who listen for the messages. In addition, hosts can be configured to query the routing switch for the information by sending Router Solicitation messages.

Some types of hosts use the Router Solicitation messages to discover their default gateway. When IRDP is enabled on the routing switch, the routing switch responds to the Router Solicitation messages. Some clients interpret this response to mean that the routing switch is the default gateway. If another router is actually the default gateway for these clients, leave IRDP disabled on the routing switch.

IRDP uses the following parameters. If you enable IRDP on individual VLAN interfaces, you can configure these parameters on an individual VLAN interface basis.

**Packet type**

The routing switch can send Router Advertisement messages as IP broadcasts or as IP multicasts addressed to IP multicast group 224.0.0.1. The default packet type is IP broadcast.

**Hold time**

Each Router Advertisement message contains a hold time value. This value specifies the maximum amount of time the host should consider an advertisement to be valid until a newer advertisement arrives. When a new advertisement arrives, the hold time is reset. The hold time is always longer than the maximum advertisement interval. Therefore, if the hold time for an advertisement expires, the host can reasonably conclude that the router interface that sent the advertisement is no longer available. The default hold time is three times the maximum message interval.

**Maximum message interval and minimum message interval**

When IRDP is enabled, the routing switch sends the Router Advertisement messages every 450-600 seconds by default. The time within this interval that the routing switch selects is random for each message and is not affected by traffic loads or other network factors. The random interval minimizes the probability that a host will receive Router Advertisement messages from other routers at the same time. The interval on each IRDP-enabled routing switch interface is independent of the interval on other IRDP-enabled interfaces. The default maximum message interval is 600 seconds. The default minimum message interval is 450 seconds.

**Preference**

If a host receives multiple Router Advertisement messages from different routers, the host selects the router that sends the message with the highest preference as the default gateway. The preference can be a number from -4294967296 to 4294967295. The default is 0.

**Enabling IRDP globally**

Enter the following command:
This command enables IRDP on the IP interfaces on all ports. Each port uses the default values for the IRDP parameters.

### Enabling IRDP on an individual VLAN interface

To enable IRDP on an individual VLAN interface and configure IRDP parameters, enter commands such as the following:

```
switch(config)# vlan 1
switch(vlan-1)# ip irdp maxadvertinterval 400
```

This example shows how to enable IRDP on a specific interface (VLAN 1) and change the maximum advertisement interval for Router Advertisement messages to 400 seconds.

**Syntax:**

```
no ip irdp [broadcast | multicast] [holdtime seconds] [maxadvertinterval seconds]  
[minadvertinterval seconds] [preference number]
```

- `broadcast | multicast`
  - Specifies the packet type the routing switch uses to send the Router Advertisement:
    - `broadcast`
      - The routing switch sends Router Advertisements as IP broadcasts.
    - `multicast`
      - The routing switch sends Router Advertisements as multicast packets addressed to IP multicast group 224.0.0.1. This is the default.

- `holdtime seconds`
  - Specifies how long a host that receives a Router Advertisement from the routing switch should consider the advertisement to be valid.
  - When a host receives a new Router Advertisement message from the routing switch, the host resets the hold time for the routing switch to the hold time specified in the new advertisement. If the hold time of an advertisement expires, the host discards the advertisement, concluding that the router interface that sent the advertisement is no longer available. The value must be greater than the value of the maxadvertinterval parameter and cannot be greater than 9000.
  - The default is three times the value of the maxadvertinterval parameter.

- `maxadvertinterval`
  - Specifies the maximum amount of time the routing switch waits between sending Router Advertisements. You can specify a value from 1 to the current value of the holdtime parameter. The default is 600 seconds.

- `minadvertinterval`
  - Specifies the minimum amount of time the routing switch can wait between sending Router Advertisements. The default is three-fourths (0.75) the value of the maxadvertinterval parameter.

  If you change the maxadvertinterval parameter, the software automatically adjusts the minadvertinterval parameter to be three-fourths the new value of the maxadvertinterval parameter. If you want to override the automatically configured value, you can specify an interval from 1 to the current value of the maxadvertinterval parameter.

- `preference number`
Specifies the IRDP preference level of this routing switch. If a host receives Router Advertisements from multiple routers, the host selects the router interface that sent the message with the highest preference as the host’s default gateway. The valid range is -4294967296 to 4294967295. The default is 0.

**Viewing IRDP information**

To display IRDP information, enter

```bash
show
ip irdp
```
from any CLI level.

**Example of output for show ip irdp**

```
switch# show ip irdp

Status and Counters - ICMP Router Discovery Protocol

  Global Status : Disabled

  VLAN Name    Status   Advertising Address  Min int (sec)  Max int (sec)  Holdtime (sec)  Preference
  ----------  ----------  ---------------  -----------  -----------  ------------  -----------
   DEFAULT_VLAN  Enabled   multicast       450         600         1800        0
    VLAN20    Enabled   multicast       450         600         1800        0
    VLAN30    Enabled   multicast       450         600         1800        0
```
The Dynamic Host Configuration Protocol (DHCP) is used for configuring hosts with IP address and other configuration parameters without user intervention. The protocol is composed of three components:

- DHCP client
- DHCP server
- DHCP relay agent

For more information, see Overview of DHCP.

**Enabling DHCP relay**

The DHCP relay function is enabled by default on a routing switch. However, if DHCP has been disabled, you can re-enable it by entering the following command at the global configuration level:

```
switch(config)# dhcp-relay
```

To disable the DHCP relay function, enter the no form of the command:

```
switch(config)# no dhcp-relay
```

**Using DHCP Option 12 to send a hostname**

This feature allows you to include the hostname in the DHCP packet sent to the DHCP server. This is disabled by default. The command must be executed from the global configuration level.

**Syntax:**
```
dhcp host-name-option
no dhcp host-name-option
```

Sends the hostname option with DHCP packets. Use the no form of the command to not include the hostname in the packet.

The maximum size of the hostname is 32 characters.

Default: disabled

**DHCP Option 12 command**

```
switch(config)# dhcp host-name-option
```

**SNMP support**

A MIB object supports enabling and disabling the DHCP Option 12 feature. It is added in the hpicfDhcpclient.mib.

- The hostname is retrieved from the MIB variable SYSNAME. Validity checks on the name include:
The name starts with a letter, ends with a letter or a digit, and can have letters, hyphens, or digits in between the first and last characters.

The maximum size supported for a hostname is 30 characters. If SYSNAME is more than 30 characters, then DHCP Option 12 will not be included in the packet.

The minimum number of characters supported for a hostname is one character. If the SYSNAME in the MIB is null, then DHCP Option 12 will not be included in the packet.

### Configuring a BOOTP/DHCP relay gateway

The DHCP relay agent selects the lowest-numbered IP address on the interface to use for DHCP messages. The DHCP server then uses this IP address when it assigns client addresses. However, this IP address may not be the same subnet as the one on which the client needs the DHCP service.

This feature provides a way to configure a gateway address for the DHCP relay agent to use for DHCP requests, rather than the DHCP relay agent automatically assigning the lowest-numbered IP address.

You must be in VLAN context to use this command, for example:

```bash
switch# config
switch(config)# vlan 1
switch(vlan-1)#
```

**Syntax:**
```
ip bootp-gateway ip-addr
```

Allows you to configure an IP address for the DHCP relay agent to use for DHCP requests. The IP address must have been configured on the interface.

**Default:** Lowest-numbered IP address

If the IP address has not already been configured on the interface (VLAN), you will see the message shown in the following example.

#### Example of trying to configure an IP address that is not on this interface (VLAN)

```bash
switch# config
switch(config)# vlan 1
switch(vlan-1)# ip bootp-gateway 10.10.10.1
The IP address 10.10.10.1 is not configured on this VLAN.
```

### Viewing the BOOTP gateway

To display the configured BOOTP gateway for an interface (VLAN) or all interfaces, enter this command. You do not need to be in VLAN context mode.

**Syntax:**
```
show dhcp-relay bootp-gateway [vlan vid]
```

Displays the configured BOOTP gateway for a specified VLAN (interface.) If a specific VLAN ID is not entered, all VLANs and their configured BOOTP gateways display.

The following example shows an IP address being assigned to a gateway for VLAN 22, and then displayed using the `show dhcp-relay bootp-gateway` command.

#### Assigning a gateway to an interface and then displaying the information

```bash
switch(vlan-22)ip bootp-gateway 12.16.18.33
switch(vlan-22)# exit
```
Operating notes

Configuring an IP helper address
To add the IP address of a DHCP server for a specified VLAN on a routing switch, enter the `ip helper-address` command at the VLAN configuration level as in the following example:

```
switch(config)# vlan 1
switch(vlan-1)# ip helper-address ip-addr
```

To remove the DHCP server helper address, enter the `no` form of the command:

```
switch(vlan-1)# no ip helper-address ip-addr
```

Operating notes

Disabling the hop count in DHCP requests
For more information, see [Hop count in DHCP requests](#).

To disable the default behavior of a DHCP relay agent so that the hop count in a DHCP client request is not increased by one at each hop when it is forwarded to a DHCP server, enter the `no dhcp-relay hop-count-increment` command at the global configuration level:

```
switch(config)# no dhcp-relay hop-count-increment
```

To reset the default function, which increases the hop count in each DHCP request forwarded to a DHCP server, enter the following command:

```
switch(config)# dhcp-relay hop-count-increment
```

Operating notes

Verifying the DHCP relay configuration

Viewing the DHCP relay setting
Use the `show config` command (or `show running` for the running-config file) to display the current DHCP relay setting.

The DHCP relay and hop count increment settings appear in the `show config` command output only if the non-default values are configured. For more information about the DHCP hop count increment, see [Hop count in DHCP requests](#).
Displaying startup configuration with DHCP relay and hop count increment disabled

```
Switch# show config
Startup configuration:
; J8697A Configuration Editor; Created on release #K.11.00
hostname "HP Switch"
cdp run
module 1 type J8702A
ip default-gateway 18.30.240.1
snmp-server community "public" Unrestricted
vlan 1
  name "DEFAULT_VLAN"
  untagged A1
  ip address 18.30.240.180 255.255.248.0
  no untagged A2-A24
  exit
  no dhcp-relay
  no dhcp-relay hop-count-increment
```

Viewing DHCP helper addresses
This command displays the list of currently configured IP Helper addresses for a specified VLAN on the switch.

Syntax:
```
show ip helper-address [vlan vlan-id]
```
Displays the IP helper addresses of DHCP servers configured for all static VLANS in the switch or on a specified VLAN, regardless of whether the DHCP relay feature is enabled. The `vlan vlan-id` parameter specifies a VLAN ID number.

Example
The following command lists the currently configured IP Helper addresses for VLAN 1.

Displaying IP helper addresses
```
switch(config)# show ip helper-address vlan 1

   IP Helper Addresses

   IP Helper Address
     -----------------  -------------
     10.28.227.97      10.29.227.53

Viewing the hop count setting
To verify the current setting for increasing the hop count in DHCP requests, enter the `show dhcp-relay` command. The current setting is displayed next to DHCP Request Hop Count Increment.

Displaying hop count status
```
switch# show dhcp-relay
Status and Counters - DHCP Relay Agent
DHCP Relay Agent Enabled : Yes
DHCP Request Hop Count Increment: Disabled
Option 82 Handle Policy : Replace
Remote ID : MAC Address

   Client Requests  Server Responses
   ---------  ---------
   Valid    Dropped  Valid    Dropped
```

Viewing the MAC address for a routing switch

To view the MAC address for a given routing switch, execute the `show system-information` command in the CLI.

Using the CLI to view the switch MAC address

```
switch(config)# show system information

Status and Counters - General System Information
System Name : switch
System Contact :
System Location :

MAC Age Time (sec) : 300
Time Zone : 0
Daylight Time Rule : None

Software revision : K.15.06.0000x   Base MAC Addr : 00110a-a50c20
ROM Version : K.15.13   Serial Number : LF713BX00E
Allow V1 Modules : No

Up Time : 32 days   Memory - Total : 128,839,680
CPU Util (%) : 0   Free : 65,802,416

IP Mgmt - Pkts Rx : 5,372,271   Packet - Total : 6750
     Pkts Tx : 298,054   Buffers Free : 5086

Lowest : 4441   Missed : 0
```

Configuring Option 82

For information on Option 82, see the sections beginning with **DHCP Option 82**.

**Syntax:**

```
```

append

Configures the switch to append an Option 82 field to the client DHCP packet. If the client packet has existing Option 82 fields assigned by another device, the new field is appended to the existing fields.

The appended Option 82 field includes the switch Circuit ID (inbound port number*) associated with the client DHCP packet and the switch Remote ID. The default switch remote ID is the MAC address of the switch on which the packet was received from the client.

To use the incoming VLAN’s IP address or the Management VLAN IP address (if configured) for the remote ID instead of the switch MAC address, use the `ip` or `mgmt-vlan` option (below.)

replace

Configures the switch to replace existing Option 82 fields in an inbound client DHCP packet with an Option 82 field for the switch.
The replacement Option 82 field includes the switch circuit ID (inbound port number*) associated with the client DHCP packet and the switch remote ID. The default switch remote ID is the MAC address of the switch on which the packet was received from the client.

To use the incoming VLAN’s IP address or the Management VLAN IP address (if configured) for the remote ID instead of the switch MAC address, use the `ip` or `mgmt-vlan` option (below.)

- **DROP**
  
  Configures the routing switch to unconditionally drop any client DHCP packet received with existing Option 82 fields. This means that such packets will not be forwarded. Use this option where access to the routing switch by untrusted clients is possible.

  If the routing switch receives a client DHCP packet without an Option 82 field, it adds an Option 82 field to the client and forwards the packet. The added Option 82 field includes the switch circuit ID (inbound port number*) associated with the client DHCP packet and the switch remote ID. The default switch remote ID is the MAC address of the switch on which the packet was received from the client.

  To use the incoming VLAN’s IP address or the Management VLAN IP address (if configured) for the remote ID instead of the switch MAC address, use the `ip` or `mgmt-vlan` option (below.)

- **KEEP**
  
  For any client DHCP packet received with existing Option 82 fields, configures the routing switch to forward the packet as-is, without replacing or adding to the existing Option 82 fields.

- **VALIDATE**
  
  Operates when the routing switch is configured with append, replace, or drop as a forwarding policy. With `validate` enabled, the routing switch applies stricter rules to an incoming Option 82 server response to determine whether to forward or drop the response. For more information, see Validation of server response packets.

**[ip | mac | mgmt-vlan]**

Specifies the remote ID suboption that the switch uses in Option 82 fields added or appended to DHCP client packets. The type of remote ID defines DHCP policy areas in the client requests sent to the DHCP server. If a remote ID suboption is not configured, the routing switch defaults to the `mac` option. See Option 82 field content.

- **ip**
  
  Specifies the IP address of the VLAN on which the client DHCP packet enters the switch.

- **mac**
  
  Specifies the routing switch’s MAC address. (The MAC address used is the same MAC address that is assigned to all VLANs configured on the routing switch.) This is the default setting.

- **mgmt-vlan**
  
  Specifies the IP address of the (optional) management VLAN configured on the routing switch. Requires that a management VLAN is already configured on the switch. If the management VLAN is multinetted, the primary IP address configured for the management VLAN is used for the remote ID. If you enter the `dhcp-relay option 82` command without specifying either `ip` or `mac`, the MAC address of the switch on which the packet was received from the client is configured as the remote ID. For information about the remote ID values used in the Option 82 field appended to client requests, see Option 82 field content.

**Example**

In the routing switch shown below, option 82 has been configured with `mgmt-vlan` for the remote ID.

```
switch(config)# dhcp-relay option 82 append mgmt-vlan
```

The resulting effect on DHCP operation for clients X, Y, and Z is shown in the following table.
Figure 12  DHCP Option 82 when using the management VLAN as the remote ID suboption

Table 34: DHCP operation for the topology in Figure DHCP Option 82 when using the management VLAN as the remote ID suboption

<table>
<thead>
<tr>
<th>Client</th>
<th>Remote ID</th>
<th>giaddr</th>
<th>DHCP server</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>10.38.10.1</td>
<td>10.39.10.1</td>
<td>A only</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>If a DHCP client is in the management VLAN, its DHCP requests can go only to a DHCP server that is also in the management VLAN. Routing to other VLANs is not allowed.</td>
</tr>
<tr>
<td>Y</td>
<td>10.38.10.1</td>
<td>10.29.10.1</td>
<td>B or C</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Clients outside of the management VLAN can send DHCP requests only to DHCP servers outside of the management VLAN. Routing to the management VLAN is not allowed.</td>
</tr>
<tr>
<td>Z</td>
<td>10.38.10.1</td>
<td>10.15.10.1</td>
<td>B or C</td>
</tr>
</tbody>
</table>

Operating notes

Overview of DHCP

The DHCP client sends broadcast request packets to the network; the DHCP servers respond with broadcast packets that offer IP parameters, such as an IP address for the client. After the client chooses the IP parameters, communication between the client and server is by unicast packets.

The routing switches provide the DHCP relay agent to enable communication from a DHCP server to DHCP clients on subnets other than the one the server resides on. The DHCP relay agent transfers DHCP messages from DHCP clients located on a subnet without a DHCP server to other subnets. It also relays answers from DHCP servers to DHCP clients.

The DHCP relay agent is transparent to both the client and the server. Neither side is aware of the communications that pass through the DHCP relay agent. As DHCP clients broadcast requests, the DHCP relay agent receives the packets and forwards them to the DHCP server. During this process, the DHCP relay agent increases the hop count by one before forwarding the DHCP message to the server. A DHCP server includes the hop count from the DHCP request that it receives in the response that it returns to the client.

DHCP packet forwarding
The DHCP relay agent on the routing switch forwards DHCP client packets to all DHCP servers that are configured in the table administrated for each VLAN.

**Unicast forwarding**

The packets are forwarded using unicast forwarding if the IP address of the DHCP server is a specific host address. The DHCP relay agent sets the destination IP address of the packet to the IP address of the DHCP server and forwards the message.

**Broadcast forwarding**

The packets are forwarded using broadcast forwarding if the IP address of the DHCP server is a subnet address or IP broadcast address (255.255.255.255). The DHCP relay agent sets the DHCP server IP address to broadcast IP address and is forwarded to all VLANs with configured IP interfaces (except the source VLAN.)

**Enabling DHCP relay operation**

For the DHCP relay agent to work on the switch, you must complete the following steps:

1. Ensure that a DHCP server is servicing the routing switch.
2. Ensure IP routing on the routing switch.
3. Ensure that there is a route from the DHCP server to the routing switch and back.
4. Configure one or more IP helper addresses for specified VLANs to forward DHCP requests to DHCP servers on other subnets.

**Hop count in DHCP requests**

When a DHCP client broadcasts requests, the DHCP relay agent in the routing switch receives the packets and forwards them to the DHCP server (on a different subnet, if necessary.) During this process, the DHCP relay agent increments the hop count before forwarding DHCP packets to the server. The DHCP server, in turn, includes the hop count from the received DHCP request in the response sent back to a DHCP client.

As a result, the DHCP client receives a non-zero hop count in the DHCP response packet. Because some legacy DHCP/BootP clients discard DHCP responses that contain a hop count greater than one, they may fail to boot up properly. Although this behavior is in compliance with RFC 1542, it prevents a legacy DHCP/BootP client from being automatically configured with a network IP address.

**DHCP Option 82**

Option 82 is called the relay agent information option and is inserted by the DHCP relay agent when forwarding client-originated DHCP packets to a DHCP server. Servers recognizing the relay agent information option may use the information to implement IP address or other parameter assignment policies. The DHCP server echoes the option back verbatim to the relay agent in server-to-client replies, and the relay agent strips the option before forwarding the reply to the client.

The relay agent information option is organized as a single DHCP option that contains one or more suboptions that convey information known by the relay agent. The initial suboptions are defined for a relay agent that is co-located in a public circuit access unit. These include a circuit ID for the incoming circuit and a remote ID that provides a trusted identifier for the remote high-speed modem.

The routing switch can operate as a DHCP relay agent to enable communication between a client and a DHCP server on a different subnet. Without Option 82, DHCP operation modifies client IP address request packets to the extent needed to forward the packets to a DHCP server. Option 82 enhances this operation by enabling the
routing switch to append an Option 82 field to such client requests. This field includes two suboptions for identifying the routing switch (by MAC address or IP address) and the routing switch port the client is using to access the network. A DHCP server with Option 82 capability can read the appended field and use this data as criteria for selecting the IP addressing it will return to the client through the usual DHCP server response packet. This operation provides several advantages over DHCP without Option 82:

- An Option 82 DHCP server can use a relay agent’s identity and client source port information to administer IP addressing policies based on client and relay agent location within the network, regardless of whether the relay agent is the client’s primary relay agent or a secondary agent.
- A routing switch operating as a primary Option 82 relay agent for DHCP clients requesting an IP address can enhance network access protection by blocking attempts to use an invalid Option 82 field to imitate an authorized client, or by blocking attempts to use response packets with missing or invalid Option 82 sub-options to imitate valid response packets from an authorized DHCP server.
- An Option 82 relay agent can also eliminate unnecessary broadcast traffic by forwarding an Option 82 DHCP server response only to the port on which the requesting client is connected, instead of broadcasting the DHCP response to all ports on the VLAN.

The routing switch's DHCP relay information (Option 82) feature can be used in networks where the DHCP servers are compliant with RFC 3046 Option 82 operation. DHCP servers that are not compliant with Option 82 operation ignore Option 82 fields.

Some client applications can append an Option 82 field to their DHCP requests.

It is not necessary for all relay agents on the path between a DHCP client and the server to support Option 82, and a relay agent without Option 82 should forward DHCP packets regardless of whether they include Option 82 fields. However, Option 82 relay agents should be positioned at the DHCP policy boundaries in a network to provide maximum support and security for the IP addressing policies configured in the server.

For more information, see the documentation provided with the server application.

**Option 82 server support**

To apply DHCP Option 82, the routing switch must operate in conjunction with a server that supports Option 82. (DHCP servers that do not support Option 82 typically ignore Option 82 fields.) Also, the routing switch applies Option 82 functionality only to client request packets being routed to a DHCP server. DHCP relay with Option 82 does not apply to switched (non-routed) client requests.

For information on configuring policies on a server running DHCP Option 82, see the documentation provided for that application.

*Example of a DHCP Option 82 application*
General DHCP Option 82 requirements and operation

Requirements
DHCP Option 82 operation is configured at the global config level and requires the following:

- IP routing enabled on the switch
- DHCP-relay option 82 enabled (global command level)
- Routing switch access to an Option 82 DHCP server on a different subnet than the clients requesting DHCP Option 82 support
- One IP helper address configured on each VLAN supporting DHCP clients

General DHCP-relay operation with Option 82
Typically, the first (primary) Option 82 relay agent to receive a client’s DHCP request packet appends an Option 82 field to the packet and forwards it toward the DHCP server identified by the IP helper address configured on the VLAN in which the client packet was received. Other, upstream relay agents used to forward the packet may append their own Option 82 fields, replace the Option 82 fields they find in the packet, forward the packet without adding another field, or drop the packet. (Intermediate next-hop routing switches without Option 82 capability can be used to forward—route—client request packets with Option 82 fields.) Response packets from an Option 82 server are routed back to the primary relay agent (routing switch) and include an IP addressing assignment for the requesting client and an exact copy of the Option 82 data the server received with the client request. The relay agent strips off the Option 82 data and forwards the response packet out the port indicated in the response as the Circuit ID (client access port.) Under certain validation conditions described later in this section, a relay agent detecting invalid Option 82 data in a response packet may drop the packet.

Example of DHCP Option 82 Operation in a Network with a Non-Compliant Relay Agent

Option 82 field content
The remote ID and circuit ID subfields comprise the Option 82 field a relay agent appends to client requests. A DHCP server configured to apply a different IP addressing policy to different areas of a network uses the values in these subfields to determine which DHCP policy to apply to a given client request.

Remote ID
This configurable subfield identifies a policy area that comprises either the routing switch as a whole (by using the routing switch MAC address) or an individual VLAN configured on the routing switch (by using the IP address of the VLAN receiving the client request.)
- Use the IP address option if the server will apply different IP addressing policies to DHCP client requests from ports in different VLANs on the same routing switch.
- Use the management VLAN option if a management VLAN is configured and you want all DHCP clients on the routing switch to use the same IP address. (This is useful if you are applying the same IP addressing policy to DHCP client requests from ports in different VLANs on the same routing switch.) Configuring this option means the management VLAN’s IP address appears in the remote ID subfield of all DHCP requests originating with clients connected to the routing switch, regardless of the VLAN on which the requests originate.
- Use the MAC address option if, on a given routing switch, it does not matter to the DHCP server which VLAN is the source of a client request (that is, use the MAC address option if the IP addressing policies supported by the target DHCP server do not distinguish between client requests from ports in different VLANs in the same routing switch.)

**Circuit ID**

This nonconfigurable subfield identifies the port number of the physical port through which the routing switch received a given DHCP client request and is necessary to identify if you want to configure an Option 82 DHCP server to use the Circuit ID to select a DHCP policy to assign to clients connected to the port. This number is the identity of the inbound port. On fixed-port switches, the port number used for the circuit ID is always the same as the physical port number shown on the front of the switch. On chassis switches, where a dedicated, sequential block of internal port numbers are reserved for each slot, regardless of whether a slot is occupied, the circuit ID for a given port is the sequential index number for that port position in the slot. (To view the index number assignments for ports in the routing switch, use the `walkmib ifname` command.)

**Using `walkmib` to determine the circuit ID for a port on a chassis**

For example, the circuit ID for port B11 on a switch is "35", as shown in the following example.

```
Switch# walkmib ifname
ifName.1 = A1
ifName.2 = A2
ifName.3 = A3
ifName.4 = A4
ifName.25 = B1
ifName.26 = B2
ifName.27 = B3
ifName.28 = B4
ifName.29 = B5
ifName.30 = B6
ifName.31 = B7
ifName.32 = B8
ifName.33 = B9
ifName.34 = B10
ifName.35 = B11
ifName.36 = B12
ifName.37 = B13
ifName.38 = B14
ifName.39 = B15
ifName.40 = B16
ifName.41 = B17
ifName.42 = B18
ifName.43 = B19
```

For example, suppose you want port 10 on a given relay agent to support no more than five DHCP clients simultaneously. You can configure the server to allow only five IP addressing assignments at any one time for the circuit ID (port) and remote ID (MAC address) corresponding to port 10 on the selected relay agent.
Similarly, if you want to define specific ranges of addresses for clients on different ports in the same VLAN, you can configure the server with the range of IP addresses allowed for each circuit ID (port) associated with the remote ID (IP address) for the selected VLAN.

**Forwarding policies**

DHCP Option 82 on switches offers four forwarding policies, with an optional validation of server responses for three of the policy types (append, replace, or drop.)

Configuration options for managing DHCP client request packets:

<table>
<thead>
<tr>
<th>Option 82 configuration</th>
<th>DHCP client request packet inbound to the routing switch</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Packet has no Option 82 field</td>
<td>Packet includes an Option 82 field</td>
</tr>
<tr>
<td>Append</td>
<td>Append an Option 82 field</td>
<td>Append</td>
</tr>
<tr>
<td></td>
<td></td>
<td>allows the most detail in defining DHCP policy boundaries. For example, where the path from a client to the DHCP Option 82 server includes multiple relay agents with Option 82 capability, each relay agent can define a DHCP policy boundary and append its own Option 82 field to the client request packet. The server can then determine in detail the agent hops the packet took, and can be configured with a policy appropriate for any policy boundary on the path.</td>
</tr>
<tr>
<td>Keep</td>
<td>Append an Option 82 field</td>
<td>If the relay agent receives a client request that already has one or more Option 82 fields, keep causes the relay agent to retain such fields and forward the request without adding another Option 82 field. But if the incoming client request does not already have any Option 82 fields, the relay agent appends an Option 82 field before forwarding the request. Some applications for keep include:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- The DHCP server does not support multiple Option 82 packets in a client request, and there are multiple Option 82 relay agents in the path to the server.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- The unusual case where DHCP clients in the network add their own Option 82 fields to their request packets</td>
</tr>
</tbody>
</table>

In networks with multiple relay agents between a client and an Option 82 server, append can be used only if the server supports multiple Option 82 fields in a client request. If the server supports only one Option 82 field in a request, consider using the keep option.
### Multiple Option 82 relay agents in a client request path

Where the client is one router hop away from the DHCP server, only the Option 82 field from the first (and only) relay agent is used to determine the policy boundary for the server response. Where there are multiple Option 82 router hops between the client and the server, you can use different configuration options on different relay agents to achieve the results you want. This includes configuring the relay agents so that the client request arrives at the DHCP client request packet inbound to the routing switch

<table>
<thead>
<tr>
<th>Option 82 configuration</th>
<th>DHCP client request packet inbound to the routing switch</th>
<th>Packet includes an Option 82 field</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Packet has no Option 82 field</td>
<td>and you do not want any additional fields added by relay agents. This policy does not include the validate option (described in the next section) and allows forwarding of all server response packets arriving inbound on the routing switch (except those without a primary relay agent identifier.)</td>
</tr>
<tr>
<td>Replace</td>
<td>Append an Option 82 field</td>
<td>Replace replaces any existing Option 82 fields from downstream relay agents (and/or the originating client) with an Option 82 field for the current relay agent. Some applications for replace include:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- The relay agent is located at a point in the network that is a DHCP policy boundary, and you want to replace any Option 82 fields appended by down-stream devices with an Option 82 field from the relay agent at the boundary. (This eliminates downstream Option 82 fields you do not want the server to use when determining which IP addressing policy to apply to a client request.)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- In applications where the routing switch is the primary relay agent for clients that may append their own Option 82 field, you can use replace to delete these fields if you do not want them included in client requests reaching the server.</td>
</tr>
<tr>
<td>Drop</td>
<td>Append an Option 82 field</td>
<td>Drop causes the routing switch to drop an inbound client request with an Option 82 field already appended. If no Option 82 fields are present, drop causes the routing switch to add an Option 82 field and forward the request. As a general guideline, configure drop on relay agents at the edge of a network, where an inbound client request with an appended Option 82 field may be unauthorized, a security risk, or for some other reason, should not be allowed.</td>
</tr>
</tbody>
</table>
server with either one Option 82 field or multiple fields. (Using multiple Option 82 fields assumes that the server supports multiple fields and is configured to assign IP addressing policies based on the content of multiple fields.)

Example configured to allow only the primary relay agent to contribute an Option 82 field

The above combination allows for detection and dropping of client requests with spurious Option 82 fields. If none are found, the drop policy on the first relay agent adds an Option 82 field, which is then kept unchanged over the next two relay agent hops ("B" and "C"). The server can then enforce an IP addressing policy based on the Option 82 field generated by the edge relay agent ("A"). In this example, the DHCP policy boundary is at relay agent 1.

Example configured to allow multiple relay agents to contribute an Option 82 field

This is an enhancement of the previous example. In this case, each hop for an accepted client request adds a new Option 82 field to the request. A DHCP server capable of using multiple Option 82 fields can be configured to use this approach to keep a more detailed control over leased IP addresses. In this example, the primary DHCP policy boundary is at relay agent "A," but more global policy boundaries can exist at relay agents "B" and "C."

Example allowing only an upstream relay agent to contribute an Option 82 field

Like the first example, above, this configuration drops client requests with spurious Option 82 fields from clients on the edge relay agent. However, in this case, only the Option 82 field from the last relay agent is retained for use by the DHCP server. In this case the DHCP policy boundary is at relay agent "C." In the previous two examples the boundary was with relay "A."

Validation of server response packets

A valid Option 82 server response to a client request packet includes a copy of the Option 82 fields the server received with the request. With validation disabled, most variations of Option 82 information are allowed, and the corresponding server response packets are forwarded.

Server response validation is an option you can specify when configuring Option 82 DHCP for append, replace, or drop Operation. See Forwarding policies. Enabling validation on the routing switch can enhance protection against DHCP server responses that are either from untrusted sources or are carrying invalid Option 82 information.

With validation enabled, the relay agent applies stricter rules to variations in the Option 82 fields of incoming server responses to determine whether to forward the response to a downstream device or to drop the response due to invalid (or missing) Option 82 information. The following table describes relay agent management of DHCP server responses with optional validation enabled and disabled.
Table 35: Relay agent management of DHCP server response packets.

<table>
<thead>
<tr>
<th>Response packet content</th>
<th>Option 82 configuration</th>
<th>Validation enabled on the relay agent</th>
<th>Validation disabled (the default)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Valid DHCP server response packet without an Option 82 field.</td>
<td>append, replace, or drop(^1)</td>
<td>Drop the server response packet.</td>
<td>Forward server response packet to a downstream device.</td>
</tr>
<tr>
<td></td>
<td>keep(^2)</td>
<td>Forward server response packet to a downstream device.</td>
<td>Forward server response packet to a downstream device.</td>
</tr>
<tr>
<td>The server response packet carries data indicating a given routing switch is the primary relay agent for the original client request, but the associated Option 82 field in the response contains a remote ID and circuit ID combination that did not originate with the given relay agent.</td>
<td>append</td>
<td>Drop the server response packet.</td>
<td>Forward server response packet to a downstream device.</td>
</tr>
<tr>
<td></td>
<td>replace or drop(^1)</td>
<td>Drop the server response packet.</td>
<td>Drop the server response packet.</td>
</tr>
<tr>
<td></td>
<td>keep(^2)</td>
<td>Forward server response packet to a downstream device.</td>
<td>Forward server response packet to a downstream device.</td>
</tr>
<tr>
<td>The server response packet carries data indicating a given routing switch is the primary relay agent for the original client request, but the associated Option 82 field in the response contains a Remote ID that did not originate with the relay agent.</td>
<td>append</td>
<td>Drop the server response packet.</td>
<td>Forward server response packet to a downstream device.</td>
</tr>
<tr>
<td></td>
<td>replace or drop(^1)</td>
<td>Drop the server response packet.</td>
<td>Drop the server response packet.</td>
</tr>
<tr>
<td></td>
<td>keep(^2)</td>
<td>Forward server response packet to a downstream device.</td>
<td>Forward server response packet to a downstream device.</td>
</tr>
<tr>
<td>All other server response packets (^3)</td>
<td>append, keep(^2), replace, or drop(^1)</td>
<td>Forward server response packet to a downstream device.</td>
<td>Forward server response packet to a downstream device.</td>
</tr>
</tbody>
</table>

\(^1\)Drop is the recommended choice because it protects against an unauthorized client inserting its own Option 82 field for an incoming request.

\(^2\)A routing switch with DHCP Option 82 enabled with the keep option forwards all DHCP server response packets except those that are not valid for either Option 82 DHCP operation (compliant with RFC 3046) or DHCP operation without Option 82 support (compliant with RFC 2131.)

\(^3\)A routing switch with DHCP Option 82 enabled drops an inbound server response packet if the packet does not have any device identified as the primary relay agent (giaddr=null; see RFC 2131.)

**Multinetted VLANs**

On a multinetted VLAN, each interface can form an Option 82 policy boundary within that VLAN if the routing switch is configured to use IP for the remote ID suboption. That is, if the routing switch is configured with IP as the remote ID option and a DHCP client request packet is received on a multinetted VLAN, the IP address used in
the Option 82 field will identify the subnet on which the packet was received instead of the IP address for the VLAN. This enables an Option 82 DHCP server to support more narrowly defined DHCP policy boundaries instead of defining the boundaries at the VLAN or whole routing switch levels. If the MAC address option (the default) is configured instead, the routing switch MAC address will be used regardless of which subnet was the source of the client request. (The MAC address is the same for all VLANs configured on the routing switch.) All request packets from DHCP clients in the different subnets in the VLAN must be able to reach any DHCP server identified by the IP helper addresses configured on that VLAN.
For introductory information about user datagram protocol (UDP), see UDP broadcast forwarding on page 278.

**Configuring and enabling UDP broadcast forwarding**

To configure and enable UDP broadcast forwarding on the switch:

1. Enable routing.
2. Globally enable UDP broadcast forwarding.
3. On a per-VLAN basis, configure a forwarding address and UDP port type for each type of incoming UDP broadcast you want routed to other VLANs.

**Globally enabling UDP broadcast forwarding**

*Syntax:*

```
no ip udp-bcast-forward
```

Enables or disables UDP broadcast forwarding on the routing switch. Routing must be enabled before executing this command.

Using the `no` form of this command disables any `ip forward protocol udp` commands configured in VLANs on the switch.

Default: Disabled

**Configuring UDP broadcast forwarding on individual VLANs**

This command routes an inbound UDP broadcast packet received from a client on the VLAN to the unicast or broadcast address configured for the UDP port type.

*Syntax:*

```
no ip forward-protocol udp ip-address [port-number | port-name]
```

Used in a VLAN context to configure or remove a server or broadcast address and its associated UDP port number. You can configure a maximum of 16 `forward-protocol udp` assignments in a given VLAN. The switch allows a total of 256 `forward-protocol udp` assignments across all VLANs.

You can configure UDP broadcast forwarding addresses regardless of whether UDP broadcast forwarding is globally enabled on the switch. However, the feature does not operate unless globally enabled.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
</table>
| ip-address | This can be either of the following:  
  - The unicast address of a destination server on another subnet. For example: 15.75.10.43.  
  - The broadcast address of the subnet on which a destination server operates. For example, the following address directs broadcasts to all hosts in the 15.75.11.0 subnet: 15.75.11.255. |
### Parameter Description

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>NOTE</strong></td>
<td>The subnet mask for a forwarded UDP packet is the same as the subnet mask for the VLAN (or subnet on a multinetted VLAN) on which the UDP broadcast packet was received from a client.</td>
</tr>
</tbody>
</table>

| udp-port# | Any UDP port number corresponding to a UDP application supported on a device at the specified unicast address or in the subnet at the specified broadcast address. For more information on UDP port numbers, refer to [TCP/UDP port number ranges on page 277](#). |

<table>
<thead>
<tr>
<th>port-name</th>
<th>Allows use of common names for certain well-known UDP port numbers. You can type in the specific name instead of having to recall the corresponding number:</th>
</tr>
</thead>
<tbody>
<tr>
<td>dns</td>
<td>Domain name service (53)</td>
</tr>
<tr>
<td>ntp</td>
<td>Network time protocol (123)</td>
</tr>
<tr>
<td>netbios-ns</td>
<td>NetBIOS name service (137)</td>
</tr>
<tr>
<td>netbios-dgm</td>
<td>NetBIOS datagram service (138)</td>
</tr>
<tr>
<td>radius</td>
<td>Remote authentication dial-in user service (1812)</td>
</tr>
<tr>
<td>radius-old</td>
<td>Remote authentication dial-in user service (1645)</td>
</tr>
<tr>
<td>rip</td>
<td>Routing information protocol (520)</td>
</tr>
<tr>
<td>snmp</td>
<td>Simple network management protocol (161)</td>
</tr>
<tr>
<td>snmp-trap</td>
<td>Simple network management protocol (162)</td>
</tr>
<tr>
<td>tftp</td>
<td>Trivial file transfer protocol (69)</td>
</tr>
<tr>
<td>timep</td>
<td>Time protocol (37)</td>
</tr>
</tbody>
</table>

**Example**

The following command configures the routing switch to forward UDP broadcasts from a client on VLAN 1 for a time protocol server:

```
switch(vlan-1)# ip forward-protocol udp 15.75.11.155 timep
```

**Viewing the current IP forward-protocol configuration**
Syntax:
show ip forward-protocol [vlan vid]
Displays the current status of UDP broadcast forwarding and lists the UDP forwarding addresses configured on all static VLANS in the switch or on a specific VLAN.

Example:

Displaying global IP forward-protocol status and configuration
This example shows the global display showing UDP broadcast forwarding status and configured forwarding addresses for inbound UDP broadcast traffic for all VLANS configured on the routing switch.

```
switch(config)# show ip forward-protocol

IP Forwarder Addresses

UDP Broadcast Forwarding: Disabled

VLAN: 1
IP Forwarder Addresses UDP Port
------------------------- -------
15.75.11.43   37
15.75.11.255  53
15.75.12.255 1813

VLAN: 2
IP Forwarder Addresses UDP Port
------------------------- -------
15.75.12.255 1812
```

Displaying IP forward-protocol status and per-VLAN configuration
This example shows the display of UDP broadcast forwarding status and the configured forwarding addresses for inbound UDP broadcast traffic on VLAN 1.

```
switch(config)# show ip forward-protocol vlan 1

IP Forwarder Addresses

UDP Broadcast Forwarding: Disabled

IP Forwarder Addresses UDP Port
------------------------- -------
15.75.11.43   37
15.75.11.255  53
15.75.12.255 1813
```

Operating notes for UDP broadcast forwarding

Maximum number of entries
The number of UDP broadcast entries and IP helper addresses combined can be up to 16 per VLAN, with an overall maximum of 2046 on the switch. (IP helper addresses are used with the switch's DHCP relay operation.) For example, if VLAN 1 has 2 IP helper addresses configured, you can add up to 14 UDP forwarding entries in the same VLAN.

TCP/UDP port number ranges
There are three ranges:

- Well-known ports: 0 to 1023
- Registered ports: 1024 to 49151
- Dynamic and/or private ports: 49152 to 65535

For more information, including a listing of UDP/TCP port numbers, go to the Internet Assigned Numbers Authority (IANA) website at:

http://www.iana.org

Then click on:

Protocol Number Assignment Services
P (Under "Directory of General Assigned Numbers" heading)
Port Numbers

Messages related to UDP broadcast forwarding

<table>
<thead>
<tr>
<th>Message</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>udp-bcast-forward: IP Routing support must be enabled first.</td>
<td>Appears in the CLI if an attempt to enable UDP broadcast forwarding has been made without IP routing being enabled first. Enable IP routing, then enable UDP broadcast forwarding.</td>
</tr>
<tr>
<td>UDP broadcast forwarder feature enabled</td>
<td>UDP broadcast forwarding has been globally enabled on the router. Appears in the Event Log and, if configured, in SNMP traps.</td>
</tr>
<tr>
<td>UDP broadcast forwarder feature disabled</td>
<td>UDP broadcast forwarding has been globally disabled on the router. This action does not prevent you from configuring UDP broadcast forwarding addresses, but does prevent UDP broadcast forwarding operation. Appears in the Event Log and, if configured, in SNMP traps.</td>
</tr>
<tr>
<td>UDP broadcast forwarder must be disabled first.</td>
<td>Appears in the CLI if you attempt to disable routing while UDP forwarding is enabled on the switch.</td>
</tr>
</tbody>
</table>

UDP broadcast forwarding

Some applications rely on client requests sent as limited IP broadcasts addressed to a UDP application port. If a server for the application receives such a broadcast, the server can reply to the client. Since typical router behavior, by default, does not allow broadcast forwarding, a client’s UDP broadcast requests cannot reach a target server on a different subnet unless the router is configured to forward client UDP broadcasts to that server.

A switch with routing enabled includes optional per-VLAN UDP broadcast forwarding that allows up to 256 server and/or subnet entries on the switch (16 entries per-VLAN.) If an entry for a particular UDP port number is configured on a VLAN, and an inbound UDP broadcast packet with that port number is received on the VLAN, the switch routes the packet to the appropriate subnet. (Each entry can designate either a single device or a single subnet. The switch ignores any entry that designates multiple subnets.)
The number of UDP broadcast forwarding entries supported is affected by the number of IP helper addresses configured to support DHCP relay. See Operating notes for UDP broadcast forwarding on page 277.

A UDP forwarding entry includes the desired UDP port number and can be either an IP unicast address or an IP subnet broadcast address for the subnet the server is in. Thus, an incoming UDP packet carrying the configured port number will be:

- Forwarded to a specific host if a unicast server address is configured for that port number.
- Broadcast on the appropriate destination subnet if a subnet address is configured for that port number.

A UDP forwarding entry for a particular UDP port number is always configured in a specific VLAN and applies only to client UDP broadcast requests received inbound on that VLAN. If the VLAN includes multiple subnets, the entry applies to client broadcasts with that port number from any subnet in the VLAN.

For example, VLAN 1 (15.75.10.1) is configured to forward inbound UDP packets as shown in the following table.

Table 36: Example of a UDP packet-forwarding environment

<table>
<thead>
<tr>
<th>Interface</th>
<th>IP address</th>
<th>Subnet mMask</th>
<th>Forwarding address</th>
<th>UDP port</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>VLAN 1</td>
<td>15.75.10.1</td>
<td>255.255.255.0</td>
<td>15.75.11.43, 15.75.11.255, 15.75.12.255</td>
<td>1188, 1812, 1813</td>
<td>Unicast address for forwarding inbound UDP packets with UDP port 1188 to a specific device on VLAN 2. Broadcast address for forwarding inbound UDP packets with UDP port 1812 to any device in the 15.75.11.0 network. Broadcast address for forwarding inbound UDP packets with UDP port 1813 to any device in the 15.75.12.0 network.</td>
</tr>
<tr>
<td>VLAN 2</td>
<td>15.75.11.1</td>
<td>255.255.255.0</td>
<td>None</td>
<td>N/A</td>
<td>Destination VLAN for UDP 1188 broadcasts from clients on VLAN 1. The device identified in the unicast forwarding address configured in VLAN 1 must be on this VLAN. Also the destination VLAN for UDP 1812 from clients on VLAN 1.</td>
</tr>
<tr>
<td>VLAN 3</td>
<td>15.75.12.1</td>
<td>255.255.255.0</td>
<td>None</td>
<td>N/A</td>
<td>Destination VLAN for UDP 1813 broadcasts from clients on VLAN 1.</td>
</tr>
</tbody>
</table>
If an IP server or subnet entry is invalid, a switch will not try to forward UDP packets to the configured device or subnet address.

**Subnet masking for UDP forwarding addresses**

The subnet mask for a UDP forwarding address is the same as the mask applied to the subnet on which the inbound UDP broadcast packet is received. To forward inbound UDP broadcast packets as limited broadcasts to other subnets, use the broadcast address that covers the subnet you want to reach. For example, if VLAN 1 has an IP address of 15.75.10.1/24 (15.75.10.1 255.255.255.0), you can configure the following unicast and limited broadcast addresses for UDP packet forwarding to subnet 15.75.11.0:

<table>
<thead>
<tr>
<th>Forwarding destination type</th>
<th>IP address</th>
</tr>
</thead>
<tbody>
<tr>
<td>UDP unicast to a single device in the 15.75.11.0 subnet</td>
<td>15.75.11.X</td>
</tr>
<tr>
<td>UDP broadcast to subnet 15.75.11.0</td>
<td>15.75.11.255</td>
</tr>
</tbody>
</table>
VRRP overview

In many networks, edge devices are often configured to send packets to a statically configured default router. If this router becomes unavailable, the devices that use it as their first-hop router become isolated from the network. Virtual Router Redundancy Protocol (VRRP) uses dynamic failover to ensure the availability of an end node’s default router. This is done by assigning the IP address used as the default route to a “virtual router” or VR. The VR includes:

- An owner router assigned to forward traffic designated for the virtual router (If the owner is forwarding traffic for the VR, it is the main router router for that VR.)
- One or more prioritized backup routers (If a backup is forwarding traffic for the VR, it has replaced the owner as the main router router for that VR.)

This redundancy provides a backup for gateway IP addresses (first-hop routers) so that if a VR’s main router router becomes unavailable, the traffic it supports will be transferred to a backup router without major delays or operator intervention. This operation can eliminate single-point-of-failure problems and provide dynamic failover (and failback) support. As long as one physical router in a VR configuration is available, the IP addresses assigned to the VR are always available, and the edge devices can send packets to these IP addresses without interruption.

Advantages to using VRRP include:

- Minimizing failover time and bandwidth overhead if a primary router becomes unavailable.
- Minimizing service disruptions during a failover.
- Providing backup for a load-balanced routing solution.
- Addressing failover problems at the router level instead of on the network edge.
- Avoiding the need to make configuration changes in the end nodes if a gateway router fails.
- Eliminating the need for router discovery protocols to support failover operation.

Both VRRPv2 and VRRPv3 are supported. IPv4 VRs can be configured for both version 2 and version 3. IPv6 VRs can only be configured for version 3.

For more information, see General operation.

Configuring VRRP

Enabling VRRP in the global configuration context

VRRP can be configured regardless of the global VRRP configuration status. However, enabling a VR and running VRRP requires enabling it in the global configuration context.

Syntax:

```
no router vrrp[IPv4][IPv6] enable|disable
```
Enables or disables VRRP operation in the global configuration context. For IPv4, IP routing must be enabled before enabling VRRP on the router. For IPv6, IPv6 unicast-routing must be enabled before enabling VRRP on the router. Disabling global VRRP halts VRRP operation on the router, but does not affect the current VRRP configuration. Enabling or disabling VRRP generates an Event Log message.

**Note:** This command has been revised from the prior `router vrrp enable` command.

To display the current global VRRP configuration, use `show vrrp config global`.

Default: Disabled

**Syntax:**
```
no router vrrp traps
```

Enables or disables SNMP trap generation for the following events:

- **New main router**
  Indicates that the sending router has transitioned to 'main router' state.

- **Authentication Failure**
  Indicates that a VRRP packet has been received from a router whose authentication key or authentication type conflicts with this router's authentication key or authentication type.

**This feature assumes the snmp-server host command has been used to configure a trap receiver. If a VRRP packet is received with an authentication type other than 0 (zero, that is, no authentication), the packet is dropped.**

Default: Enabled

**Example**

**Enabling and displaying the global VRRP configuration**

The following commands enable VRRP at the global configuration level and then display the current global VRRP configuration:

```
switch(config)# router vrrp
switch(config)# show vrrp config global

VRRP Global Configuration Information
  VRRP Enabled : Yes
  Traps Enabled : Yes
```

**Creating a VR and entering the VR context**

**Syntax:**
```
no vrrp [ipv6] vrid 1-255
```

Used in the VLAN interface context to create a virtual router (VR) instance and to enter the context of the new VR. It is also used to enter the context of an existing VR, and is the method used for accessing a VR for configuration purposes. You can configure up to 32 VRs on a multinetted VLAN (16 for the 2930F switch).

The VLAN interface must be IP enabled for IPv4, and IPv6 enabled for IPv6.

**Example:**

To create VR 1 in VLAN 10 and enter the VR context, execute the following command:

```
switch(vlan-10)# vrrp vrid 1
switch(vlan-10-vrid-1)#
```

**Selecting a Version of VRRP**
The version command is configured in the IPv4 VR context. IPv4 virtual routers support VRRPv2 and VRRPv3. IPv6 virtual routers support only VRRPv3. The default is version 2. The show running-config command will display the version only when it is set at the non-default value of version 3 for IPv4.

If an update is performed from a software version that only supported VRRPv2, all the IPv4 VRs remain in VRRPv2 mode.

**Syntax:**
version 2|3

**Example:**
switch(vlan-10-vrid-1)# version 3

## Configuring a VR instance on a VLAN interface

This section describes the configuration and activation commands available in the VR context.

### Configuring a virtual IP address (VIP) in a VR

The VIP must be the same for the owner and all backups on the same network or subnet in a VR. The decision about the VIP being an owner or backup is based on the whether the VIP is a configured IP address or not. If the VIP is a configured IP address, it becomes the owner and the priority becomes 255. For backup VRs, the priority is between 1-254.

**Syntax:**
- virtual-ip-address ipv4-addr
- virtual-ip-address ipv6-addr

Used in a VR context of a VLAN to assign an IPv4 address for IPv4, or an IPv6 address for IPv6, to a VR instance.

This command had a subnet option in prior versions. This is not needed now because the subnet is provided when the VLAN IP address is configured.

**For an owner**

The VIP must be one of the IP addresses configured on the VLAN interface for that VR.

**For a backup**

The VIP must match the VIP for the owner.

The owner and the backups using a given VIP must all belong to the same network or subnet.

For IPv6, you must configure the link-local address before you are able to configure the global IPv6 address. In the owner VR, this is the link-local address of the interface. The VR can be enabled only after configuring the first virtual IP address. Additional IP addresses can be configured without disabling the VR.

When removing virtual IP addresses, when the last virtual IP address is removed, the VR state is changed to disabled. A warning message is displayed.

---

**NOTE**

If the link-local address is changed, the VR configuration is removed. When an IP address is removed, the virtual IP address in that subnet is removed.

Only IPv6 addresses are processed when in IPv6 context.

Default: None
Example

If VLAN 10 on router "A" is configured with an IP address of 10.10.10.1/24 and VR 1, and you want router "A" to operate as the owner for this VR, the VIP of the owner in VR 1 on router "A" is also 10.10.10.1/24. On router "B," which will operate as a backup for VR 1, VLAN 10 is configured (in the same network) with an IP address of 10.10.10.15/24. However, because the backup must use the same VIP as the owner, the VIP for the backup configured on router "B" for VR 1 is also 10.10.10.1/24.

VIP assignment for owner and backup

Reconfiguring the priority for a backup

When you configure a backup in a VR, it is given a default priority of 100. This command is intended for use where it is necessary to establish a precedence among the backup routers on the same network or subnet in a given VR.

Syntax:

priority 1-254

Used in a VR context of a VLAN where the router is configured as a backup. This command changes the backup's priority and is used to establish the precedence of a backup where there are multiple backups belonging to the same network or subnet.

An owner is automatically assigned the highest priority, 255, which cannot be changed unless the owner status is reconfigured to backup.

Default: 100; Range: 1 - 254, where 1 is the lowest precedence

Changing VR advertisement interval and source IP address

Syntax:

vrrp vrid vrid-num advertise-interval 1-40
vrrp ipv6 vrid vrid-num advertise-interval 1-40

- When a VRRP router is operating as main router, this value specifies the interval at which the router sends an advertisement notifying the other VRRP routers on the network or subnet that a main router is active.
- When a VRRP router is operating as a backup, it uses this value to calculate a main router down interval (3 x advt-interval.)

Default for IPv4: 1 second; range: 1-40 seconds
Default for IPv6: 1 second; range: 1-40 seconds

For information on advertisements and advertisement intervals, see Function of the VRRP Advertisement.
All VRRP routers belonging to the same VR must be configured with the same advertisement interval. As required in RFC 3768, if a locally configured advertisement interval does not match the interval received in an inbound VRRP packet, the VR drops that packet.

Syntax:
```
primary-ip-address [ip-address | lowest]
```

For IPv4 only. IPv6 does not have a primary-ip-address option; the primary IP address is the link-local address of the real interface over which the packet is transmitted.

Specifies the VIP to designate as the source for VRRP advertisements from the VR. If there is only one VIP configured on the VR, the default setting \( \text{lowest} \) is sufficient. Where there are multiple VIPs in the same VR and you want to designate an advertisement source other than the lowest IP Address, use this command.

For an owner VR, the primary IP address must be one of the VIPs configured on the VR.

For a backup VR, the primary IP address must be in the same subnet as one of the VIPs configured on the VR. In addition, the primary IP address for a backup VR must be one of the IP addresses configured on the VLAN on which the VR is configured.

Executed in VRID context.

Default: lowest

It is common in VRRP applications to have only one VIP per VR. In such cases, the protocol uses that address as the source IP address for VRRP advertisements, and it is not necessary to specify an address.

**Configuring preempt mode on VRRP backup routers**

This command applies to VRRP backup routers only and is used to minimize network disruption caused by unnecessary preemption of the main router operation among backup routers. It is executed in VRID context.

Syntax:
```
no preempt-mode
```

Disables or re-enables preempt mode. In the default mode, a backup router coming up with a higher priority than another backup that is currently operating as main router will take over the main router function.

Using the `no` form of the command disables this operation, thus preventing the higher-priority backup from taking over the main router operation from a lower-priority backup.

This command does not prevent an owner router from resuming the main router function after recovering from being unavailable.

Default: Enabled

**Enabling or disabling VRRP operation on a VR**

Syntax:
```
no enable
```

Enabling or disabling a VR enables or disables dynamic VRRP operation on that VR.

Default: Disabled

**Dynamically changing the priority of the VR**
You can configure tracked interfaces or VLANs on the backup router only.

## Configuring track interface

**Syntax**

```
no track interface [port-list|trunk-list]
```

Allows you to specify a port or port list, or trunk or trunk list, that will be tracked by this virtual router. If the port or trunk is down, the virtual router switches to the router specified by the priority value. The command is executed in VRID instance context.

**Example:**

```
switch(config)# vlan 25
switch(vlan-25)# vrid 1
switch(vlan-25-vrid-1)# track interface 10-12, Trk1
```

## Configuring track VLAN

The VR's operating VLAN cannot be configured as a tracking VLAN for that VR.

**Syntax:**

```
no track vlan <vlan-id>range
```

Allows you to specify a VLAN or range of VLANs that will be tracked by this virtual router. If the VLAN is down, or if the VLAN or IP address has been deleted, the virtual router switches to the router specified by the priority value. The command is executed in VRID instance context.

**Example:**

```
switch(config)# vlan 25
switch(vlan-25)# vrid 1
switch(vlan-25-vrid-1)# track vlan 10 24-26
```

When the first tracked port or tracked VLAN comes up after being down, the VR waits for the pre-empt delay time before it tries to take control back. The VR resumes being a backup with its configured priority as soon as the first tracked entity is up.

The behavior of the VR is not affected by any tracked entities until after the expiration of the preempt delay time. However, if while waiting for the preempt delay time to expire, a main router goes down, the VR tries to take control of the virtual IP.

## Removing all tracked entities

**Syntax:**

```
no track
```

Allows you to remove tracking for all configured track entities (ports, trunks, and VLANs.) The command is executed in VRID instance context.
Example:

```plaintext
switch(vlan-25-vrid-1)# no track
```

### Viewing VRRP tracked entities

You can display the VRRP tracked entities by entering the command shown in this example.

#### Example showing results of `show vrrp tracked-entities` command

```plaintext
switch(vlan-25-vrid-1)# show vrrp tracked-entities

VRRP Tracked entities

<table>
<thead>
<tr>
<th>VLAN</th>
<th>VR ID</th>
<th>Type</th>
<th>ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>25</td>
<td>1</td>
<td>port</td>
<td>7</td>
</tr>
<tr>
<td>25</td>
<td>1</td>
<td>port</td>
<td>12</td>
</tr>
<tr>
<td>25</td>
<td>1</td>
<td>port</td>
<td>13</td>
</tr>
<tr>
<td>25</td>
<td>1</td>
<td>port</td>
<td>14</td>
</tr>
<tr>
<td>25</td>
<td>1</td>
<td>vlan</td>
<td>1</td>
</tr>
</tbody>
</table>
```

### Forcing the backup VR operating as main router to relinquish ownership of the VR instance

**Syntax:**

```plaintext
failover [with-monitoring]
```

The command is executed in VRID instance context.

### Forcing the backup VR to take ownership of the VR instance

Failback is disabled on the owner VR; it can be executed only on the backup VR. Failback can occur only on a VR on which `failover` or `failover with-monitoring` has been executed.

**Syntax:**

```plaintext
failback
```

This command takes effect only if the backup VR instance has a higher priority than the current owner, which is normal VRRP router behavior. The command is executed in VRID instance context.

### Configuring the Authentication Data Field

The `null-auth-compatibility` command is used to allow inter-operation with other switches that expect authentication data fields in VRRPv3 packets for IPv6. It is configured in the VR context. This command allows compatibility with some Comware switches.

**Syntax:**

```plaintext
no null-auth-compatibility
```

When this command is enabled, authentication data is appended at the end of an IPv6 VRRP packet that is being sent. The authentication data consists of 8 bytes of zeros.
Pinging the virtual IP of a backup router

Enabling the response to a ping request
The backup router can be enabled to respond to pings using the following command. For more information, see Pinging the virtual IP of a backup router.

Syntax:
Parameters router vrrp virtual-ip-ping

Enables or disables the response to a ping request for the switch. When enabled, all VRs that are not owners and are not explicitly disabled (see virtual-ip-ping enabled command) respond to ping requests sent to the VIP when the backup VR is acting as main router.

By default, response to virtual IP ping is enabled only when the VR ID is created and enabled.

Enabling the response to ping requests

```
router1# config
router1(config)# ip routing
router1Router1(config)# router vrrp
router1(config)# router vrrp virtual-ip-ping
```

Controlling ping responses

Syntax:
Parameters virtual-ip-ping enabled

Enables or disables the response to a ping request to a specific VR. The command applies to all VIPs on the VR. Must be executed in VRRP context (vlan vid vrrp vrid vrid)

The VR should be configured as a backup.

Default: Enabled

Disabling a response to ping requests to a VIP

```
switch-Router1(config)# ip routing
switch-Router1(config)# router vrrp
switch-Router1(config)# router vrrp virtual-ip-ping
switch-Router1(config)# vlan 2 vrrp vrid 1
switch-Router1(vlan-2-vrid-1)# virtual-ip-address 10.0.202.87
switch-Router1(vlan-2-vrid-1)# no virtual-ip-ping enable

switch-Router1(vlan-2-vrid-1)# enable
switch-Router1(vlan-2-vrid-1)# exit
switch-Router1(vlan-2)# exit
switch-Router1(config)#
```

Viewing VRRP ping information

Display IPv4 global VRRP configuration information by entering the show vrrp config global command.
Display IPv6 global VRRP configuration information by entering the show vrrp ipv6 config global command.
### Example of VRRP global configuration information

```bash
switch(config)# show vrrp config global

VRPP Global Configuration Information

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>VRRP Enabled</td>
<td>Yes</td>
</tr>
<tr>
<td>Traps Enabled</td>
<td>Yes</td>
</tr>
<tr>
<td>Virtual Routers Respond to Ping Requests</td>
<td>Yes</td>
</tr>
<tr>
<td>Virtual Nonstop enabled</td>
<td>No</td>
</tr>
</tbody>
</table>
```

Use the `show vrrp` command to display information about VRRP global statistics.

### VRRP IPv4 global statistics information

```bash
switch(config)# show vrrp

VRPP Global Statistics Information

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>VRRP Enabled</td>
<td>Yes</td>
</tr>
<tr>
<td>Invalid VRID Pkts Rx</td>
<td>0</td>
</tr>
<tr>
<td>Checksum Error Pkts Rx</td>
<td>0</td>
</tr>
<tr>
<td>Bad Version Pkts Rx</td>
<td>0</td>
</tr>
<tr>
<td>Virtual Routers Respond To Ping Requests</td>
<td>Yes</td>
</tr>
</tbody>
</table>

VRPP Virtual Router Statistics Information

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vlan ID</td>
<td>2</td>
</tr>
<tr>
<td>Virtual Router ID</td>
<td>1</td>
</tr>
<tr>
<td>Protocol Version</td>
<td>2</td>
</tr>
<tr>
<td>State</td>
<td>Main Router</td>
</tr>
<tr>
<td>Up Time</td>
<td>25 secs</td>
</tr>
<tr>
<td>Virtual MAC Address</td>
<td>00005e-000101</td>
</tr>
<tr>
<td>Main Router's IP Address</td>
<td>10.0.102.87</td>
</tr>
<tr>
<td>Associated IP Addr Count</td>
<td>1</td>
</tr>
<tr>
<td>Near Failovers</td>
<td>0</td>
</tr>
<tr>
<td>Advertise Pkts Rx</td>
<td>0</td>
</tr>
<tr>
<td>Become Main Router</td>
<td>1</td>
</tr>
<tr>
<td>Zero Priority Rx</td>
<td>0</td>
</tr>
<tr>
<td>Zero Priority Tx</td>
<td>0</td>
</tr>
<tr>
<td>Bad Length Pkts</td>
<td>0</td>
</tr>
<tr>
<td>Bad Type Pkts</td>
<td>0</td>
</tr>
<tr>
<td>Mismatched Interval Pkts</td>
<td>0</td>
</tr>
<tr>
<td>Mismatched Addr List Pkts</td>
<td>0</td>
</tr>
<tr>
<td>Mismatched IP TTL Pkts</td>
<td>0</td>
</tr>
<tr>
<td>Mismatched Auth Type Pkts</td>
<td>0</td>
</tr>
</tbody>
</table>
```

### Example

### VRRP IPv6 global statistics information

```bash
switch# show vrrp ipv6 statistics

VRPP Global Statistics Information

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>VRRP Enabled</td>
<td>Yes</td>
</tr>
<tr>
<td>Invalid VRID Pkts Rx</td>
<td>0</td>
</tr>
<tr>
<td>Checksum Error Pkts Rx</td>
<td>0</td>
</tr>
<tr>
<td>Bad Version Pkts Rx</td>
<td>0</td>
</tr>
<tr>
<td>Virtual Routers Respond To Ping Requests</td>
<td>Yes</td>
</tr>
</tbody>
</table>

VRPP Virtual Router Statistics Information

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>VLAN ID</td>
<td>10</td>
</tr>
<tr>
<td>Virtual Router ID</td>
<td>1</td>
</tr>
<tr>
<td>Protocol Version</td>
<td>3</td>
</tr>
<tr>
<td>State</td>
<td>Main Router</td>
</tr>
</tbody>
</table>
```
Display VRRP configuration information using the `show vrrp config` command.

Example

**VRRP IPv4 configuration display showing VIP ping status**

```
switch# show vrrp config

VRRP Global Configuration Information

VRRP Enabled : Yes
Traps Enabled : Yes
Virtual Routers Respond To Ping Requests : Yes
VRRP Nonstop Enabled : No

VRRP Virtual Router Configuration Information

VLAN ID 2
Virtual Router ID : 1

Administrative Status [Disabled] : Enabled
Mode [Uninitialized] : backup
Priority [100] : 150
Advertisement Interval [1] : 1
Preempt Mode [True] : True
Preempt Delay Time [0] : 0
Respond To Virtual IP Ping Requests [Yes] : Yes
Null authentication compatibility [False] : False
Primary IP Address : Lowest

IP Address
----------
10.0.202.87
```

Example

**VRRP IPv6 configuration display showing VIP ping status**

```
switch# show vrrp ipv6 config

VRRP Global Configuration Information

VRRP Enabled : Yes
Traps Enabled : Yes
Virtual Routers Respond To Ping Requests : No
VRRP Nonstop Enabled : No

VRRP Virtual Router Configuration Information

VLAN ID : 10
Virtual Router ID : 10
```
Example

**VRRP IPv4 configuration for a VLAN and VRID**

This example displays the ping response status for a specific VLAN and VRID.

```plaintext
switch(config)# show vrrp vlan 2 vrid 1 config

VRRP Virtual Router Configuration Information

  VLAN ID : 2
  Virtual Router ID : 1

  Administrative Status [Disabled] : Enabled
  Mode [Uninitialized] : backup
  Priority [100] : 150
  Advertisement Interval [1] : 1
  Preempt Mode [True] : True
  Preempt Delay Time [0] : 0
  Respond To Virtual IP Ping Requests [Yes] : Yes
  RFC2338 authentication compatibility : True

  IPv6 Address
  ----------------
  fe80::216:b9ff:fed1:5280
```

Example

**VRRP IPv6 configuration for a VLAN and VRID**

```plaintext
switch# show vrrp ipv6 vlan 10 vrid 4 config

VRRP Virtual Router Configuration Information

  VLAN ID : 10
  Virtual Router ID : 1
  Administrative Status [Disabled] : Enabled
  Mode [Uninitialized] : Owner
  Priority [100] : 255
  Advertisement Interval [1] : 1
  Preempt Mode [True] : True
  Preempt Delay Time [0] : 0
  Respond To Virtual IP Ping Requests [Yes] : Yes

  IPv6 Address
  ----------------
  fe80::216:b9ff:fed1:5280
```
Example

IP route information

This example shows the gateway information for IP routes. A designation of "reject" means that the IP traffic for that route is discarded. Null/reject routes are added when a backup VR becomes a main router and takes ownership of all VIPs. The null/reject route ensures that routed packets to the VIP are not forwarded through it. When virtual-ip-ping is enabled, the ping packets to the VIP are responded to with ping replies.

```
switch(config)# show ip route

+-----------------+-----------------+-------+-----------------+--------+--------+
| Destination     | Gateway          | VLAN  | Type            | Sub-Type| Metric |
+-----------------+-----------------+-------+-----------------+--------+--------+
| 10.0.0.0/16     | DEFAULT_VLAN    | 1     | connected       | 1       | 0      |
| 10.0.202.87/32  | reject          |       | static          | 1       | 1      |
| 127.0.0.0/8     | reject          |       | static          | 0       | 0      |
| 127.0.0.1/32    | lo0             |       | connected       | 1       | 0      |
+-----------------+-----------------+-------+-----------------+--------+--------+
```

Operational notes

- Jumbo frames are supported if they have been enabled for that VLAN. The VIP responds to ping requests if they are not fragmented and are not larger than the MTU.
- Fragmented packets are not supported. All fragmented packets sent to a VIP are dropped and no response or error is sent.
- All packets with IP options are dropped. Any ping options will work as long as they do not change to IP options.
- ICMP requests other than echo requests are not supported.
- If there are errors in packets sent to a VIP, for example, "TTL Invalid," no ICMP error packet is sent.

Specifying the time a router waits before taking control of the VIP

For more information, see Using the Pre-empt Delay Timer (PDT).

Syntax:
```
Parameters preempt-delay-time 1-1600
```

Allows you to specify a time in seconds that this router will wait before taking control of the VIP and beginning to route packets. You can configure the timer on VRRP owner and backup routers.

If you have configured the preempt delay time (PDT) with a non-zero value, you must use the no form of the command to change it to 0 (zero.)

Default: 0 (zero) seconds.

Viewing VRRP configuration data

Viewing the VRRP global configuration

Syntax:
```
show vrrp config global
```

Displays the configuration state for the global VRRP configuration and VRRP trap generation.

Output showing the default global VRRP configuration (IPv4/IPv6)
switch(config)# show vrrp config

VRRP Global Configuration Information

<table>
<thead>
<tr>
<th>VRRP Enabled</th>
<th>No</th>
</tr>
</thead>
<tbody>
<tr>
<td>Traps Enabled</td>
<td>Yes</td>
</tr>
<tr>
<td>Virtual Routers Respond to Ping Requests</td>
<td>No</td>
</tr>
<tr>
<td>VRRP Nonstop Enabled</td>
<td>No</td>
</tr>
</tbody>
</table>

**Viewing all VR configurations on the router**

**Syntax:**
show vrrp config

Displays the configuration for the global VRRP configuration and all VRs configured on the router.

**VRRP IPv4 configuration listing with two owner VRs configured**

This example lists output indicating two owner VRs configured on the router.

switch(config)# show vrrp config

VRRP Global Configuration Information

<table>
<thead>
<tr>
<th>VRRP Enabled</th>
<th>Yes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Traps Enabled</td>
<td>Yes</td>
</tr>
<tr>
<td>Virtual Routers Respond To Ping Requests</td>
<td>Yes</td>
</tr>
<tr>
<td>VRRP Nonstop Enabled</td>
<td>No</td>
</tr>
</tbody>
</table>

VRRP Virtual Router Configuration Information

VLAN ID : 2
Virtual Router ID : 10

Administrative Status [Disabled] : Disabled
Mode [Uninitialized] : Owner
Priority [100] : 255
Advertisement Interval [1] : 1
Preempt Mode [True] : True
Preempt Delay Time [0] : 0
Respond To Virtual IP Ping Requests [Yes] : Yes
Null authentication compatibility [False] : False
Primary IP Address : Lowest

IP Address
----------
10.10.10.100

VRRP Virtual Router Configuration Information

VLAN ID : 20
Virtual Router ID : 20

Administrative Status [Disabled] : Enabled
Mode [Uninitialized] : Owner
Priority [100] : 255
Advertisement Interval [1] : 1
Preempt Mode [True] : True
Preempt Delay Time [0] : 0
Respond To Virtual IP Ping Requests [Yes] : Yes
Null authentication compatibility [False] : False
Primary IP Address : Lowest
IP Address
----------
10.10.20.100

Example

VRRP IPv6 Configuration Listing

switch# show vrrp ipv6 config

VRRP Global Configuration Information

VRRP Enabled : Yes
Traps Enabled : Yes
Virtual Routers Respond To Ping Requests : No
VRRP Nonstop Enabled : No

VRRP Virtual Router Configuration Information

VLAN ID : 10
Virtual Router ID : 10
Administrative Status [Disabled] : Enabled
Mode [Uninitialized] : Owner
Priority [100] : 255
Advertisement Interval [1] : 1
Preempt Mode [True] : True
Preempt Delay Time [0] : 0
Respond To Virtual IP Ping Requests [Yes] : Yes
RFC2338 authentication compatibility : True

IPv6 Address
----------
fe80::216:b9ff:fed1:5280

Viewing a specific VR configuration

Syntax:
show vrrp vlan 23 vrid 10 config

Displays the configuration for a specific VR in a specific VLAN.

Displaying the IPv4 configuration for a specific VR

The following command displays the configuration of a VR identified as VR 10 in VLAN 23:

switch(config)# show vrrp vlan 23 vrid 10 config

VRRP Virtual Router Configuration Information

Vlan ID : 23
Virtual Router ID : 10

Administrative Status [Disabled] : Disabled
Mode [Uninitialized] : Owner
Priority [100] : 255
Advertisement Interval [1] : 1
Preempt Mode [True] : True
Preempt Delay Time [0] : 0
Respond to Virtual IP Ping Requests [Yes] : Yes
Displaying the IPv6 configuration for a specific VR

```
switch# show vrrp ipv6 vlan 10 vrid 4 config

VRRP Virtual Router Configuration Information

VLAN ID : 10
Virtual Router ID : 1
Administrative Status [Disabled] : Enabled
Mode [Unitialized] : Owner
Priority [100] : 255
Advertisement Interval [1] : 1
Preempt Mode [True] : True
Preempt Delay Time [0] : 0
Respond To Virtual IP Ping Requests [Yes] : Yes

IPv6 Address
---------------------
fe80::216:b9ff:fed1:5280
```

Viewing VRRP statistics data

All command outputs shown in this section assume that VRRP is enabled at the global configuration level. If global VRRP is disabled, these commands produce the following output:

Example

```
show vrrp statistics
c
```

Viewing global VRRP statistics only

Syntax:
```
show vrrp statistics global
show vrrp ipv6 statistics global
```

Displays the global VRRP statistics for the router:

- VRRP Enabled [Yes/No]
- Invalid VRID Pkts Rx: VRRP packets received for a VRID that is not configured on the specific VLAN of the VRRP router.
- Checksum Error Pkts Rx: VRRP packets received with a bad checksum
- Bad Version Pkts Rx: VRRP advertisement packets received with a version number other than 2 or 3.
- Virtual Routes Respond to Ping Requests [Yes/No]

Example
Global VRRP statistics output

The output is the same for IPv4 and IPv6.

```
switch(config)# show vrrp statistics global

VRRP Global Statistics Information

  VRRP Enabled : Yes
  Invalid VRID Pkts Rx : 0
  Checksum Error Pkts Rx : 0
  Bad Version Pkts Rx : 0
  Virtual Routers Respond to Ping Requests : No
```

Viewing statistics for all VRRP instances on the router

**Syntax:**
show vrrp [statistics]
show vrrp ipv6 statistics

Displays the following VRRP statistics:

- Global VRRP statistics for the router
- VRRP statistics for all VRs configured on the router:

**State**
Indicates whether the router is a backup or the current main router of the VR.

**Uptime**
The amount of time the router has been up since the last reboot.

**Virtual MAC Address**
The virtual MAC address for the VR instance.

**main router's IP Address**
The IP address used as the source IP address in the last advertisement packet received from the VR main router. If this VR is the main router, this is the primary IP address of the VR. If the VR is disabled, this value appears as 0.0.0.0 for IPv4, and 0:0:0:0:0:ffff:0:0 for IPv6.

**Associated IP Address Count**
Number of VIPs.

**Advertise Packets Rx**
The number of VRRP main router advertisements the VR has received from other VRRP routers since the last reboot.

**Zero Priority Tx**
The number of VRRP advertisement packets received with the priority field set to 0 (zero.)

**Bad Length Pkts**
The number of VRRp packets received with missing fields of information.

**Mismatched Interval Pkts**
The number of VRRP packets received from other routers (since the last reboot) with an advertisement interval that is different from the interval configured on the current VR. VRRP packets received with an interval mismatch are dropped.

**Mismatched IP TTL Pkts**
The number of VRRP packets received with the IP TTL field not set to 255. Such packets are dropped.

**Near Failovers**
Tracks the occurrence of "near failovers" on the backup VRRP routers. This makes visible any difficulties the VRRP routers are having receiving the "heartbeat" advertisement from the main router. A "near failover" is one that is within one missed VRRP advertisement packet of beginning the main router determination process.

**Become main router**

The number of times the VR has become the main router since the last reboot.

**Zero Priority Tx**

The number of VRRP advertisement packets sent with the priority field set to 0 (zero.)

**Bad Type Pkts**

The number of VRRP packets received with packet type not equal to 1 (that is, not an advertisement packet.)

**Mismatched Addr List Pkts**

The number of VRRP packets received wherein the list of VIPs does not match the locally configured VIPs for a VR.

**Mismatched Auth Type Pkts**

The number of VRRP packets received with the authentication type not equal to 0 (zero, which is no authentication.)

---

The commands `show vrrp` and `show vrrp statistics` result in the same output.

**Example**

**Output for `show vrrp` command includes global and VR statistics**

The following output shows the VRRP statistics on a router having one VR (VR 1 in VLAN 10) configured.

```
switch(config)# show vrrp

VRRP Global Statistics Information

  VRRP Enabled            : Yes
  Invalid VRID Pkts Rx    : 0
  Checksum Error Pkts Rx  : 0
  Bad Version Pkts Rx     : 0
  Virtual Routers Respond to Ping Requests : No

VRRP Virtual Router Statistics Information

  Vlan ID                : 10
  Virtual Router ID      : 1
  Protocol Version       : 2
  State                  : Main Router
  Up Time                : 31 mins
  Virtual MAC Address    : 00005e-000101
  Main Router's IP Address : 10.10.10.2
  Associated IP Addr Count : 1
  Near Failovers         : 0
  Advertise Pkts Rx      : 1213
  Become Main Router     : 2
  Zero Priority Rx       : 0
  Zero Priority Tx       : 0
  Bad Length Pkts        : 0
  Bad Type Pkts          : 0
  Mismatched Interval Pkts : 0
  Mismatched Addr List Pkts : 0
  Mismatched IP TTL Pkts : 0
  Mismatched Auth Type Pkts : 0
```

**Output for `show vrrp ipv6 statistics` command includes global and IPv6 VR statistics**

```
switch# show vrrp ipv6 statistics

VRRP Global Statistics Information
```
VRRP Enabled: Yes
Invalid VRID Pkts Rx: 0
Checksum Error Pkts Rx: 0
Bad Version Pkts Rx: 0
Virtual Routers Respond To Ping Requests: Yes

VRRP Virtual Router Statistics Information
VLAN ID: 10
Virtual Router ID: 1
Protocol Version: 3
State: Main Router
Up Time: 26 mins
Virtual MAC Address: 00005e-000101
Main Router's IP Address: 2130::21
Associated IP Addr Count: 1
Near Failovers: 0
Advertise Pkts Rx: 0
Become Main Router: 1
Zero Priority Rx: 0
Zero Priority Tx: 0
Bad Length Pkts: 0
Bad Type Pkts: 0
Mismatched Interval Pkts: 0
Mismatched Addr List Pkts: 0
Mismatched IP TTL Pkts: 0
Mismatched Auth Type Pkts: 0

Viewing statistics for all VRRP instances in a VLAN

Syntax:
show vrrp vlan vid [statistics]

Displays the VRRP statistics for all VRs configured on the specified VLAN.
The actual statistics data per VR is the same as for the show vrrp [statistics] command shown on pages A-24 and Viewing statistics for all VRRP instances on the router.

Note that show vrrp vlan vid and
show vrrp vlan vid statistics
produce the same output.

Displaying IPv4 statistics for all VRs in a VLAN

In the following example, there is one VR configured in VLAN 10.

switch(config)# show vrrp vlan 10

VRRP Virtual Router Statistics Information

Vlan ID: 10
Virtual Router ID: 10
Protocol Version: 2
State: Main Router
Up Time: 6 mins
Virtual MAC Address: 00005e-00010a
Main Router's IP Address: 10.10.10.1
Associated IP Addr Count: 1
Near Failovers: 0
Advertise Pkts Rx: 1
Become Main Router: 1
Zero Priority Rx: 0
Zero Priority Tx: 0
Bad Length Pkts: 0
Bad Type Pkts: 0
Mismatched Interval Pkts: 0
Mismatched Addr List Pkts: 0
Mismatched IP TTL Pkts: 0
Mismatched Auth Type Pkts: 0

Displaying IPv6 statistics for all VRs in a VLAN
switch# show vrrp ipv6 vlan 10 statistics

VRRP Virtual Router Statistics Information
VLAN ID : 10
Virtual Router ID : 1
Protocol Version : 3
State : Main Router
Up Time : 26 mins
Virtual MAC Address : 00005e-000101
Main Router's IP Address : 2130::21
Associated IP Addr Count : 1
Near Failovers : 0
Advertise Pkts Rx : 0
Become Main Router : 1
Zero Priority Rx : 0
Zero Priority Tx : 0
Bad Length Pkts : 0
Bad Type Pkts : 0
Mismatched Interval Pkts : 0
Mismatched Addr List Pkts : 0
Mismatched IP TTL Pkts : 0
Mismatched Auth Type Pkts : 0

Viewing statistics for a specific VRRP instance

Syntax:
show vrrp vlan vid vrid 1-255 [statistics]
show vrrp ipv6 vlan vid vrid 1-255 statistics

Displays the VRRP statistics for a specific VR configured on a specific VLAN.
The actual statistics data per VR is the same as for the show vrrp [statistics] command.

Note that
show vrrp vlan vid vrid 1 - 255

and

show vrrp
vlan vid vrid 1 - 255

statistics produce the same output.

Viewing the "near-failovers" statistic

The "near failovers" statistic tracks occurrences of near failovers on the backup VRRP routers. This makes visible any difficulties the VRRP routers are having receiving the "heartbeat" advertisement from the main router. (A "near failover" is one that is within one missed VRRP advertisement packet of beginning the main router determination process.)

The show vrrp or show vrrp ipv6 statistics command displays this statistic.

Example Output

The show vrrp command with statistics

Near Failovers statistic displayed is shown in bold below.

switch(config)# show vrrp

VRRP Global Statistics Information
VRRP Enabled : Yes
Invalid VRID Pkts Rx : 0
Checksum Error Pkts Rx : 0
Bad Version Pkts Rx : 0
Virtual Routers Respond to Ping Requests : No

VRRP Virtual Router Statistics Information

<table>
<thead>
<tr>
<th>Vlan ID</th>
<th>22</th>
</tr>
</thead>
<tbody>
<tr>
<td>Virtual Router ID</td>
<td>1</td>
</tr>
<tr>
<td>Protocol Version</td>
<td>2</td>
</tr>
<tr>
<td>State</td>
<td>Initialize</td>
</tr>
<tr>
<td>Up Time</td>
<td>64 mins</td>
</tr>
<tr>
<td>Virtual MAC Address</td>
<td>00005e-000101</td>
</tr>
<tr>
<td>Main Router's IP Address</td>
<td>10.10.20.2</td>
</tr>
</tbody>
</table>

Associated IP Addr Count : 1  
Near Failovers : 0  
Advertise Pkts Rx : 0  
Become Main Router : 0  
Zero Priority Rx : 0  
Zero Priority Tx : 0  
Bad Length Pkts : 0  
Bad Type Pkts : 0  
Mismatched Interval Pkts : 0  
Mismatched Addr List Pkts : 0  
Mismatched IP TTL Pkts : 0  
Mismatched Auth Type Pkts : 0

The `show vrrp ipv6 statistics` command

switch# show vrrp ipv6 statistics

VRRP Global Statistics Information

VRRP Enabled : Yes
Invalid VRID Pkts Rx : 0
Checksum Error Pkts Rx : 0
Bad Version Pkts Rx : 0
Virtual Routers Respond To Ping Requests : Yes

VRRP Virtual Router Statistics Information

<table>
<thead>
<tr>
<th>VLAN ID</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>Virtual Router ID</td>
<td>1</td>
</tr>
<tr>
<td>Protocol Version</td>
<td>3</td>
</tr>
<tr>
<td>State</td>
<td>Main Router</td>
</tr>
<tr>
<td>Up Time</td>
<td>26 mins</td>
</tr>
<tr>
<td>Virtual MAC Address</td>
<td>00005e-000101</td>
</tr>
<tr>
<td>Main Router's IP Address</td>
<td>2130::21</td>
</tr>
</tbody>
</table>

Associated IP Addr Count : 1  
Near Failovers : 0  
Advertise Pkts Rx : 0  
Become Main Router : 1  
Zero Priority Rx : 0  
Zero Priority Tx : 0  
Bad Length Pkts : 0  
Bad Type Pkts : 0  
Mismatched Interval Pkts : 0  
Mismatched Addr List Pkts : 0  
Mismatched IP TTL Pkts : 0  
Mismatched Auth Type Pkts : 0

Using the debug command with the VRRP option

The `vrrp` option with the `debug` command turns on the tracing of the incoming and outgoing VRRP packets.

Syntax:
```
Parameters debug vrrp
```

Displays VRRP debug messages.

General operation

VRRP supports router redundancy through a prioritized election process among routers configured as members of the same virtual router (VR).
On a given VLAN, a VR includes two or more member routers configured with a VIP that is also configured as a real IP address on one of the routers, plus a virtual router MAC address. The router that owns the IP address is configured to operate as the owner of the VR for traffic-forwarding purposes and by default has the highest VRRP priority in the VR. The other routers in the VR have a lower priority and are configured to operate as backups in case the owner router becomes unavailable.

The owner normally operates as the main router for a VR. But if it becomes unavailable, then a failover to a backup router belonging to the same VR occurs, and this backup becomes the current main router. If the owner recovers, a failback occurs and "main router" status reverts to the owner. (Using more than one backup provides additional redundancy" if both the owner and the highest-priority backup fail, another, lower-priority backup can take over as main router.)

- The VIP used by all VRRP routers in a VR instance is a real IP address that is also configured on the applicable VLAN interface on the VR's owner router.

- The same MAC and VIPs are included in the VRRP configuration for the owner and all backup routers belonging to the same VR and are used as the source addresses for all traffic forwarded by the VR.

The following example shows a VR on VLAN 100 supported by Router 1 (R1) and Router 2 (R2.)

*Example of using VRRP to provide redundant network access*

![Diagram of VRRP configuration](image)

As long as R1 remains available, it operates as the master. If R1 fails, then R2 takes over as master.

<table>
<thead>
<tr>
<th>VR parameter</th>
<th>Router 1 VR configuration</th>
<th>Router 2 VR configuration</th>
<th>Operation</th>
</tr>
</thead>
<tbody>
<tr>
<td>VRID (Virtual Router ID)</td>
<td>1</td>
<td>1</td>
<td>All routers in the same VR have the same VRID.</td>
</tr>
<tr>
<td>Status</td>
<td>owner</td>
<td>backup</td>
<td>One owner and one or more backups are allowed in a given VR.</td>
</tr>
</tbody>
</table>
### VR parameter | Router 1 VR configuration | Router 2 VR configuration | Operation
---|---|---|---
Virtual IP Address | 10.10.100.1 | 10.10.100.1 | The IP address configured for VLAN 100 in R1 (the owner) is also configured as the VIP for VRRP in both R1 and R2.
VR Source MAC Address | 00-00-5E-00-01-01 | | For any VR in any VLAN, this is always defined as 00-00-5E-00-01-01 - VRID and is not configurable.
Priority | 255 (Default) | 100 (Default) | The router configured as owner in any VR is automatically assigned the highest priority (255.) backup routers are assigned a default priority of 100, which can be reconfigured.

In the preceding example:

- Host "A" uses 10.10.100.1 as its next-hop gateway out of the subnet, as represented by the VR (VR 1.)
  - Router 1 (the configured owner) advertises itself as the main router in the VR supporting the gateway and:
    - "Owns" the VR's (virtual) IP address
    - Transmits ARP responses that associate the VR's VIP with the (shared) source MAC address for VR 1.
  - During normal operation, Router 1 forwards the routed traffic for host "A."
- If Router 1 fails or otherwise becomes unavailable:
  1. Router 1 advertisements of its main router status for VR 1 fail to reach Router 2 (which is the only configured backup.)
  2. After the time-out period for receiving main router advertisements expires on Router 2, the VR initiates a failover to Router 2 and it becomes the new main router of the VR.
  3. Router 2 advertises itself as the main router of the VR supporting the gateway and:
    - Takes control of the VR's (virtual) IP address
    - Begins transmitting ARP responses that associate the VR's VIP with the (shared) source MAC address for VR 1
  4. Host "A" routed traffic then moves through Router 2.
- If Router 1 again becomes available:
  1. Router 1 resumes advertising itself as the main router for the VR and sends ARP responses that associate the VR's VIP with the (shared) source MAC address for VR 1.
  2. Router 2 receives the advertisement from Router 1 and ceases to operate as the VR’s main router, and halts further transmission of its own VRRP advertisements and ARP responses related to VR 1.
  3. The VR executes a failback to Router 1 as main router, and Host "A" traffic again moves through Router 1.
Virtual router (VR)

A VR instance consists of one owner router and one or more backup routers belonging to the same network. Any VR instance exists within a specific VLAN, and all members of a given VR must belong to the same subnet. In a multinetted VLAN, multiple VRs can be configured. The owner operates as the VR’s main router unless it becomes unavailable, in which case the highest-priority backup becomes the VR’s main router.

A VR includes the following:

- VR identification (VRID) configured on all VRRP routers in the same network or, in the case of a multinetted VLAN, on all routers in the same subnet.
- Same VIP configured on each instance of the same VR.
- Status of either owner or backup configured on each instance of the same VR (on a given VR, there can be one owner and one or more backups.)
- Priority level configured on each instance of the VR (on the owner router the highest priority setting, 255, is automatically fixed; on backups, the default priority setting is 100 and is configurable.)
- VR MAC address (not configurable.)

Where a VLAN is configured with only one network (IP address), one VR is allowed in that VLAN. In a multinetted VLAN, there can be one VR per subnet, with a maximum of 32 VRs (16 for the 2930F switch) in any combination of main routers and backups.

NOTE

All routers in a given VR must belong to the same network (or subnet, in the case of a multinetted VLAN.)

Virtual IP address (VIP)

The VIP associated with a VR must be a real IP address already configured in the associated VLAN interface on the owner router in the VR. If the VIP is an IPv6 address, a link-local address must be configured before adding a global IPv6 address. Also, the owner and all other (backup) routers belonging to the VR have this IP address configured in their VRID contexts as the VIP. In General operation, 10.10.100.1 is a real IP address configured on VLAN 100 in Router 1 and is the VIP associated with VR 1.

If the configured owner in a VR becomes unavailable, it is no longer the main router for the VR and a backup router in the VR is elected to assume the role of main router, as described under Backup router.

A subnetted VLAN allows multiple VIPs. However, if there are 32 or fewer IP addresses in a VLAN interface, and you want VRRP support on multiple subnets, the recommended approach is to configure a separate VR instance for each IP address in the VLAN. In cases where VRRP support is needed for more than 32 IP addresses in the same VLAN.

Main Router

The current main router in a VR operates as the "real" or physical gateway router for the network or subnet for which a VIP is configured.

Control of Main Router Selection

Selection of the main router is controlled by the VRRP priority value configured in the VRID context of each router in the VR. The router configured as the owner in the VR is automatically assigned the highest VRRP priority (255) and, as long as it remains available, operates as the main router for the VR. The other routers belonging to the VR as backups are assigned the default priority value (100) and can be reconfigured to any priority value between 1 and 254, inclusive. If the current main router becomes unavailable, the protocol uses the priority values configured on the other, available routers in the VR to select another router in the VR to take over the main router function.
**Function of the VRRP Advertisement**

The current main router sends periodic advertisements to inform the other routers in the VR of its operational status. If the backup VRs fail to receive a main router advertisement within the timeout interval, the current main router is assumed to be unavailable and a new main router is elected from the existing backups. The timeout interval for a VR is three times the advertisement interval configured on the VRs in the network or subnet. In the default VRRP configuration, the advertisement interval is one second and the resulting timeout interval is three seconds.

---

All VRRP routers belonging to the same VR must be configured with the same advertisement interval. As required in RFC 3768, if a locally configured advertisement interval does not match the interval received in an inbound VRRP packet, the VR drops that packet.

---

Most IPv6 host configurations learn the default gateway IPv6 address using router advertisements. The VR that becomes the main router sends router advertisements for its virtual IP address.

**Owner router**

An owner router for a VR is the default main router for the VR and operates as the owner for all subnets included in the VR. The VRRP priority on an owner router is always 255 (the highest.)

---

On a multi netted VLAN where multiple subnets are configured in the same VR, the router must be either the owner for all subnets in the VR or a backup for all subnets in the VR.

---

**Backup router**

There must be at least one backup router. A given VR instance on a backup router must be configured with the same VIP as the owner for that VR (and both routers must belong to the same network or subnet.) Router 2 in General operation illustrates this point.

**VR Priority Operation**

In a backup router's VR configuration, the virtual router priority defaults to 100. (The priority for the configured owner is automatically set to the highest value: 255.) In a VR where there are two or more backup routers, the priority settings can be reconfigured to define the order in which backups are reassigned as main router in the event of a failover from the owner.

**Preempt mode**

Where multiple backup routers exist in a VR, if the current main router fails and the highest-priority backup is not available, VRRP selects the next-highest priority backup to operate as main router. If the highest-priority backup later becomes available, it preempts the lower-priority backup and takes over the main router function. If you do not want a backup router to have this preemptive ability on a particular VR, you can disable this operation with the no preempt-mode command. (Preempt mode applies only to VRRP routers configured as backups.)

**Virtual router MAC address**

When a VR instance is configured, the protocol automatically assigns a MAC address based on the standard MAC prefix for VRRP packets, plus the VRID number (as described in RFC 3768.) The first five octets form the standard MAC prefix for VRRP, and the last octet is the configured VRID. that is: 00-00-5E-00-01- VRid

For example, the virtual router MAC address for the VR in General operation is 00-00-5E-00-01-01.
VRRP and ARP for IPv4

The main router for a given VR responds to ARP requests for the VIPs with the VR’s assigned MAC address. The virtual MAC address is also used as the source MAC address for the periodic advertisements sent by the current main router.

The VRRP router responds to ARP requests for non-VIPs (IP addresses on a VLAN interface that are not configured as VIPs for any VR on that VLAN) with the system MAC address.

VRRP and neighbor discovery for IPv6

Neighbor Discovery (ND) is the IPv6 equivalent of the IPv4 ARP for layer 2 address resolution, and uses IPv6 ICMP messages to do the following:

- Determine the link-layer address of neighbors on the same VLAN interface.
- Verify that a neighbor is reachable.
- Track neighbor (local) routers.

Neighbor Discovery enables functions such as the following:

- Router and neighbor solicitations and discovery
- Detecting address changes for devices on a VLAN
- Identifying a replacement for a router or router path that has become unavailable
- Duplicate address detection (DAD)
- Router Advertisement processing
- Neighbor reachability
- Autoconfiguration of unicast addresses
- Resolution of destination addresses
- Changes to link-layer addresses.

An instance of Neighbor Discovery is triggered on a device when a new or changed IPv6 address is detected. VRRPv3 provides a faster failover to a backup router by not using standard ND procedures. A failover to a backup router can occur in approximately three seconds without any interaction with hosts and with a minimum of VRRPv3 traffic.

Duplicate address detection (DAD)

Duplicate Address Detection verifies that a configured unicast IPv6 address is unique before it is assigned to a VLAN interface. When the owner router fails, the backup VRRP router assumes the main router role. When the owner router becomes operational, DAD will fail as there is a backup VRRP router in the main router role that responds to the DAD request. To avoid this, virtual routers that are in owner mode (priority = 255) will not send DAD requests for the VLAN interface on which the owner VR is configured.

General operating rules

- IP routing (IPv4) or IPv6 unicast-routing (IPv6) must be enabled on the router before enabling VRRP.
- IP must be enabled on a VLAN before creating a VR instance on the VLAN.
- **VIP: On an owner**
  - The VIP configured in a VR instance must match one of the IP addresses configured in the VLAN interface on which the VR is configured.
  - **On a backup**
The VIP configured in a VR instance cannot be a "real" IP address configured in a VLAN interface on that router.

The VIP configured for one VR cannot be configured on another VR.

- **Before**
  - changing a router from owner to backup, or the reverse, the VIP must be removed from the configuration.
  - The priority configuration on an owner can be only 255. The priority configuration on a backup must be 254 or lower, the default being 100.
  - Advertisement intervals:
    - If a VRRP router has a different advertisement interval than a VRRP packet it receives, the router drops the packet. For this reason, the advertisement interval must be the same for the owner and all backups in the same VR.
  - A VR exists within a single VLAN interface. If the VLAN is multinetted, a separate VR can be configured within the VLAN for each subnet. A VLAN allows up to 32 VRs (16 for the 2930F switch), and the switch allows up to 2048 VRs.
  - All routers in the same VR must belong to the same network or subnet.
  - The router supports the following maximums:
    - 32 VRs (16 for the 2930F switch) per VLAN in any combination of main routers and backups
    - 512 (128 for the 2930F switch) IPv4 and IPv6 VRs in combination
    - 2046 Virtual IP addresses
    - 512 (128 for the 2930F switch) VR sessions on the switch
    - 512 (128 for the 2930F switch) VRRPv2 and VRRPv3 sessions, in any mix
    - 32 (16 for the 2930F switch) IP addresses per VR
  - Each VR uses one MAC address as described under [Virtual router MAC address](#).
  - If an IP address is deleted on a VLAN interface, one of the following occurs:
    - VR owner: If the VR uses the same IP address as a VIP, that IP address is deleted from the VR.
    - VR backup: If the VR has a VIP in the same subnet as that of the deleted IP address, that VIP will be deleted from the VR.

If the deleted VIP was the last VIP of an active VR, the VR will be deactivated. (For more on multiple, VIPs on a VR, see [Associating more than one VIP with a VR](#).

- The VRRP backup router can respond to ping requests when the virtual-ip-ping feature is enabled. For more information, see [Pinging the virtual IP of a backup router](#).

**Steps for provisioning VRRP operation**

**Basic configuration process**

This process assumes the following for VRRP operation:

- VLANs on the selected routers are already configured and IP-enabled.
- IP routing (IPv4) or IPv6 unicast-routing (IPv6) is enabled.
- The network topology allows multiple paths for routed traffic between edge devices.
1. Configure the owner for VRRP operation and a VR instance.
   1. On the router intended as the owner for a particular network or subnet, enter the global configuration context and enable VRRP:

   ```
   router vrrp ipv4 enable or
   router vrrp
   ipv6 enable
   ```

   2. Enter the desired VLAN context and configure a VR instance:

   ```
   vlan vid vrrp vrid 1 - 255 (for IPv4)
   vrrp ipv6 vrid 1-255 (for IPv6)
   ```

   This step places the CLI in the context of the specified VR.

   3. Configure the router’s real IP address for the current VLAN interface as the VIP for the VR instance.

   ```
   virtual-ip-address ipaddr
   ```

   4. Activate the owner VR instance:

   ```
   enable
   ```

   5. Inspect the configuration for the owner VR:

   ```
   show vrrp vlan vid vrid vrid-# config
   ```

   (IPv4)

   ```
   show vrrp ipv6 vlan vid vrid vrid-# config
   ```

   (IPv6)

   2. Leave the owner’s advertisement interval at its default (1 second).

   3. Configure a backup for the same VR instance as for the owner in step Configure the owner for VRRP operation and a VR instance. On the router intended as the owner for a particular network or subnet, enter the global configuration context and enable VRRP:

   ```
   router vrrp ipv4 enable or
   router vrrp ipv6 enable
   ```

   Enter the desired VLAN context and configure a VR instance:

   ```
   vlan vid vrrp vrid 1 - 255 (for IPv4)
   vrrp ipv6 vrid 1-255 (for IPv6)
   ```

   This step places the CLI in the context of the specified VR. Configure the router’s real IP address for the current VLAN interface as the VIP for the VR instance, virtual-ip-address ipaddr, activate the owner VR instance, enable, inspect the configuration for the owner VR:

   ```
   show vrrp vlan vid vrid vrid-# config (IPv4)
   show vrrp ipv6 vlan vid vrid vrid-# config (IPv6)
   ```

   1. On another router with an interface in the same network or subnet as is the owner configured in step Configure the owner for VRRP operation and a VR instance. On the router intended as the owner for a particular network or subnet, enter the global configuration context and enable VRRP:

   ```
   router vrrp ipv4 enable or
   router vrrp ipv6 enable
   ```

   Enter the desired VLAN context and configure a VR instance:

   ```
   vlan vid vrrp vrid 1 - 255 (for IPv4)
   vrrp ipv6 vrid 1-255 (for IPv6)
   ```

   This step places the CLI in the context of the specified VR. Configure the router’s real IP address for the current VLAN interface as the VIP for the VR instance, virtual-ip-address ipaddr, activate the owner VR instance, enable, inspect the configuration for the owner VR:

   ```
   show vrrp vlan vid vrid vrid-# config (IPv4)
   show vrrp ipv6 vlan vid vrid vrid-# config (IPv6)
   ```

   Enter the global configuration context and enable VRRP:

   ```
   router vrrp ipv4 enable or
   ```

   2. Configure (and enter) the same VR instance as was configured for the owner in step Configure the owner for VRRP operation and a VR instance. On the router intended as the owner for a particular network or subnet, enter the global configuration context and enable VRRP:

   ```
   router vrrp ipv4
   ```
enable router vrrp ipv6 enable

Enter the desired VLAN context and configure a VR instance:

```
vlan vid vrrp vrid 1 - 255 (for IPv4)
vrrp ipv6 vrid 1-255 (for IPv6)
```

This step places the CLI in the context of the specified VR. Configure the router’s real IP address for the current VLAN interface as the VIP for the VR instance.

```
virtual-ip-address ipaddr
```

Activate the owner VR instance:

```
enable
```

Inspect the configuration for the owner VR:

```
show vrrp vlan vid vrid vrid-# config (IPv4)
show vrrp ipv6 vlan vid vrid vrid-# config (IPv6)
```

3. Optional: If there is only one backup router, or if you want the priority among backups to be determined by the lowest IP address among the backups, leave the VR instance priority for the current backup router at the default of 100. (Applies only to the "real" IP addresses that are part of this VR—there may be other addresses on the routers that are lower—but only the interfaces participating in the VR are part of this determination.) If you want to control backup router priority by creating a numeric hierarchy among the backup routers in the VR, set the priority on each accordingly:

```
priority 1
- 254
```

4. Configure the VIP for the current VR. Use the same address as you used for the owner router’s instance of the VR.

```
virtual-ip-address ipaddr
```

5. Activate the backup VR instance:

```
enable
```

6. Inspect the configuration for the owner VR:

```
show vrrp vlan vid vrid vrid-# config
```

```
show vrrp ipv6 vlan vid vrid vrid-# config
```

(for IPv6) Leave the advertisement interval for backup routers at the default (1 second).

4. Repeat step 2 for each backup router on the same VR.

Example configuration

In VR 1, below, R1 is the owner and the current main router, and R2 is the (only) backup in the VR. If R1 becomes unavailable, VR 1 fails over to R2.

Example of a basic VRRP configuration
### VRRP configuration for Router 1 (R1) in Example of a basic VRRP configuration

```
switch(config)# router vrrp
switch(config)# vlan 10
switch(vlan-10)# vrrp vrid 1
switch(vlan-10-vrid-1)# owner
switch(vlan-10-vrid-1)# virtual-ip-address 10.10.10.1
switch(vlan-10-vrid-1)# enable

switch(vlan-10-vrid-1)# show vrrp vlan 10 vrid 1 config

VRRP Virtual Router Configuration Information

VLAN ID : 10
Virtual Router ID : 1

Administrative Status [Disabled] : Enabled
Mode [Uninitialized] : owner
Priority [100] :255
Advertisement Interval [1] : 1
Preempt Mode [True] : True
Preempt Delay time [0] : 0
Respond to Virtual IP Ping Requests [Yes] : Yes
Null authentication compatibility [False] : False
Primary IP Address : Lowest

IP Address
-----------
10.10.10.1
```
Associating more than one VIP with a VR

If a VLAN is configured with more than 32 (16 for the 2930F switch) subnets and it is necessary to apply VRRP to all of these subnets, it is necessary to associate more than one VIP with a VR.

Because a VLAN on the routers supports up to 32 (16 for the 2930F switch) VRs, applying VRRP to a higher number of subnets in the VLAN requires multiple VIPs in one or more VRRPs.

If the owner of a VR is associated with multiple VIPs, the backup routers belonging to the same VR must also be associated with the same set of VIPs. If the VIPs on the owner are not also on the backups, a misconfiguration exists. VRRP advertisement packets sent by the VR main router will be dropped by the VR backups because of a mismatch among VIPs.

Dynamically changing the priority of the VR

You can configure tracked interfaces or VLANs on the backup router only.

Failover operation

Failover operation involves handing off the VR’s control of the virtual IP to another VR. Once a failover command is issued, the VR begins sending advertisements with priority zero instead of the configured priority. When the VR detects a peer VR taking control, it releases control of the virtual IP and ceases VR operation until a failback is executed. Failover occurs on only a backup VR operating as main router.

If you specify the with-monitoring option, the VR continues to monitor the virtual IP after ceasing VR operation. If the main router VR goes down, it then retakes control of the virtual IP.

Pinging the virtual IP of a backup router

Using the Pre-empt Delay Timer (PDT)

To maintain availability of the default gateway router, the VRRP advertises a "virtual" router to the hosts. At least two other physical routers are configured to be virtual routers, but only one router provides the default router functionality at any given time. If the owner router or its VLAN goes down, the backup router takes over. When the owner router comes back on line (fail-back), it takes control of the VIP that has been assigned to it. It begins sending out VRRP advertisement packets at regular intervals. The backup router receives the VRRP advertisement packet and transitions to the backup state.

When OSPF is also enabled on the VRRP routers

When OSPF is enabled on the routers and a fail-back event occurs, the owner router immediately takes control of the VIP and provides the default gateway functionality. If OSPF has not converged, the route table in the owner router may not be completely populated. When the hosts send packets to the default gateway, the owner router may not know where to send them and packets may be dropped.

While you can run OSPF and VRRP concurrently on a router, it is best not to run VRRP with other routing protocols, such as RIP or OSPF, on the same interface or VLAN, as this can create operational issues.
Configuring the PDT

The VRRP PDT allows you to configure a period of time before the VR takes control of the VIP. It does not transition to the main router state until the timer period expires. The timer value configured should be long enough to allow OSPF convergence following OSPF updates.

The PDT is applied only during initialization of the router, that is, when the router is rebooting with the VRRP parameters present in the startup config file.

VRRP preempt mode with LACP and older devices

There can be an issue with VRRP preempt mode if an older device is the intermediate device connecting to a VRRP router and has LACP set in "enable, passive" mode. This mode is set by default on older devices, whereas it is disabled by default on later models. It is recommended that you use compatible LACP settings on devices that connect with VRRP routers on VRRP VLANs.

What occurs at startup

When the owner router comes online, it waits for the configured amount of time before taking control of the VIP. This period of time is calculated as follows:

If the value of the main router down time (3 * advertisement interval) is less than or equal to the preempt delay time, the owner router will wait until the main router down time (3 * advertisement interval) has expired.

During this waiting period, if the owner router receives a VRRP packet for its VIP from the backup router, it waits until the PDT expires before taking control of its VIP. If the owner router does not receive any VRRP packets and the main router down time expires, the owner router can take control of its VIP immediately.

If the value of the main router down time (3 * advertisement interval) is greater than the preempt delay time, the owner router will wait until the PDT expires before taking control of its VIP.

Selecting a value for the PDT

You should select the value for the PDT carefully to allow time for OSPF to populate the owner router’s route tables. The choice depends on the following:

- The OFPF router dead interval—the number of seconds the OSPF router waits to receive a hello packet before assuming its neighbor is down.
- The number of router interfaces that participate in OSPF
- The time it may take from reception of the OSPF packets to when the population of the route table is completed.

There are trade-offs between selecting a small advertisement value and a large PDT. A small advertisement value results in a faster failover to the backup router. A larger PDT value allows OSPF to converge before the owner router takes back control of its VIP.

Choosing a large PDT value (greater than the main router down time) may result in an unnecessary failover to the backup router when the VRRP routers (owner and backup) start up together. Choosing a large advertisement interval and thereby a large main router down time results in a slower failover to the backup router when the owner router fails.

Possible configuration scenarios

PDT=zero seconds

This is the default behavior. It works in the same way that VRRP works currently.
PDT is greater than or equal to the main router down time (3 times the advertisement interval)

1. An owner VR after reboot—waits for the main router down time. If the owner router does not receive a packet during this time, it becomes the main router. If it receives a VRRP advertisement from its peer during this time, it waits until the expiration of the preempt delay time before becoming the main router.

2. A backup VR after reboot—waits for the main router down time. If the backup router does not receive a packet during this time, it becomes the main router. If it receives a VRRP advertisement from its peer during this time, and it has a higher priority value than this peer, it waits until the expiration of the preempt delay time before becoming the backup.

PDT is less than the main router down time

1. Owner router—becomes the main router after expiration of the PDT.

2. Backup router—becomes the backup after expiration of the PDT if it does not receive a VRRP advertisement from a higher priority peer (or the owner.)

When the PDT is not applicable

Once the router has rebooted and is in steady state VRRP operation, the PDT is not applicable if:

- The VRRP VLAN goes down and comes back up.
- The VR is disabled and re-enabled.
- VRRP is globally disabled and then re-enabled.

Standards compliance

VRRP on the switches includes the following:

- Complies with RFC 3768 VRRP version 2.
- Complies with RFC 5798 version 3 with two exceptions—advertisement intervals below one second are not supported, and accept mode is not supported (only ping application for virtual-ip-ping).
- Complies with RFC 2787—Definitions of Managed Objects for VRRP, except for support for authentication-related values.
- Unified standard MIB RFC 6527 supports both IPv4 and IPv6.
- Private MIB hpicfVrrp3.mib is added to support new IPv6 extensions and VR-specific extensions of the deprecated, private MIB hpicfVrrp.mib. Global extensions still use hpicfVrp.mib

Operating notes

Dynamic priority change operating notes

- There are no backward compatibility issues with the VRRP dynamic priority change feature. If a VRRP router has an older firmware version that does not have the dynamic priority change feature, it will not have the needed configuration options.
- The VR’s operating VLAN cannot be configured as a tracking VLAN for that VR.
- Ports that are part of a trunk cannot be tracked.
- A port that is tracked cannot be included in a trunk.
- Trunks that are tracked cannot be removed; you are not able to remove the last port from the trunk.
- LACP (active or passive) cannot be enabled on a port that is being tracked.
- If a VLAN is removed or a port becomes unavailable, the configuration is retained and they are tracked when they become available again.
- After the owner VR relinquishes control of its IP address, that IP address becomes unavailable to all other applications and routing protocols such as RIP and OSPF.
- To avoid operational issues, Hewlett Packard Enterprise recommends that VRRP is not run on the same interface/VLAN with other routing protocols, such as RIP and OSPF.

**VRRP-Services enhancements**

Initially, VRRP virtual IP address supported ping service only. Now, VRRP will support all the IP services such as ping, Telnet, SSH, and web support.

The `virtual-ip-ping` command is removed because IP services are always enabled.

```
show vrrp
```

**Syntax**
```
show vrrp
```

**Description**
Displays VRRP global statistics information.

**Example**
```
switch#show vrrp
VRRP Global Statistics Information

VRRP Enabled : Yes
Invalid VRID Pkts Rx : 0
Checksum Error Pkts Rx : 0
Bad Version Pkts Rx : 0

VRRP Virtual Router Statistics Information

Vlan ID : 21
Virtual Router ID : 21
Protocol Version : 3
State : Main Router
Up Time : 27 hours
Virtual MAC Address : 00005e-000115
Main Router's IP Address : 21.0.0.2
Associated IP Addr Count : 1 Near Failovers : 1
Advertise Pkts Rx : 5 Become Main Router : 1
Zero Priority Rx : 0 Zero Priority Tx : 0
Bad Length Pkts : 0 Bad Type Pkts : 0
Mismatched Interval Pkts : 0 Mismatched Addr List Pkts : 0
Mismatched IP TTL Pkts : 0 Mismatched Auth Type Pkts : 0
BFD Administrative Status : Disabled
BFD Operational Status : Disabled
```

```
show vrrp config
```
Syntax

show vrrp config

Description
Displays VRRP virtual router configuration information

Example

```
switch#show vrrp config
VRRP Global Configuration Information

   VRRP Enabled             : Yes
   Traps Enabled            : Yes
   VRRP Nonstop Enabled     : No

VRRP Virtual Router Configuration Information

   VLAN ID : 21
   Virtual Router ID : 21

   Administrative Status [Disabled] : Enabled
   Mode [Uninitialized] : Backup
   Priority [100] : 252
   Advertisement Interval [1] : 1
   Preempt Mode [True] : True
   Preempt Delay Time [0] : 1
   Respond To Virtual IP Ping Requests [Yes] : Yes
   Null authentication compatibility [False] : False
   Primary IP Address : Lowest

   IP Address
   ------------
   21.0.0.10
```

show vrrp statistics

Syntax

show vrrp statistics

Description
Displays VRRP virtual router statistics information.

Example

```
switch#show vrrp statistics
VRRP Global Statistics Information

   VRRP Enabled             : Yes
   Invalid VRID Pkts Rx    : 0
   Checksum Error Pkts Rx  : 0
   Bad Version Pkts Rx     : 0
```
show vrrp ipv6 config

Syntax

show vrrp ipv6 config

Description

Displays VRRP virtual router configuration information.

Example

switch# show vrrp ipv6 config
VRRP Global Configuration Information

  VRRP Enabled : Yes
  Traps Enabled : Yes
  VRRP Nonstop Enabled : No

VRRP Virtual Router Configuration Information

  VLAN ID : 21
  Virtual Router ID : 101

  Administrative Status [Disabled] : Enabled
  Mode [Uninitialized] : Owner
  Priority [100] : 255
  Advertisement Interval [1] : 1
  Preempt Mode [True] : True
  Preempt Delay Time [0] : 0
  Respond To Virtual IP Ping Requests [Yes] : Yes
  Null authentication compatibility [False] : False

  IPv6 Address
  ------------
  fe80::a21d:48ff:fe8f:a800

show vrrp ipv6 statistics

Syntax
show vrrp ipv6 statistics

Description
Displays VRRP virtual router statistics information.

Example

switch# show vrrp ipv6 statistics
VRPP Global Statistics Information
VRRP Enabled : Yes
Invalid VRID Pkts Rx : 0
Checksum Error Pkts Rx : 0
Bad Version Pkts Rx : 0

VRRP Virtual Router Statistics Information
Vlan ID : 21
Virtual Router ID : 101
Protocol Version : 3
State : Main Router
Up Time : 5 mins
Virtual MAC Address : 00005e-000265
Main Router's IP Address : fe80::a21d:48ff:fe8f:a800
Associated IP Addr Count : 1 Near Failovers : 0
Advertise Pkts Rx : 0 Become Main Router : 1
Zero Priority Rx : 0 Zero Priority Tx : 0
Bad Length Pkts : 0 Bad Type Pkts : 0
Mismatched Interval Pkts : 0 Mismatched Addr List Pkts : 0
Mismatched IP TTL Pkts : 0 Mismatched Auth Type Pkts : 0
BFD Administrative Status : Disabled
BFD Operational Status : Disabled

VRRP-Show command enhancement

show vrrp summary

Syntax
show vrrp summary

Description
Shows useful information for all VRRP IPv4 instances on the device.

Example

switch# show vrrp summary
Vlan VRID Pri Up Time Owner Pre State Main Router Addr Virtual Addr
-------- ---- ----- ------- ----- ------- ----------------- 
10 11 11 2 mins Backup True Main Router 11.1.1.2 11.1.1.9
10 123 202 2 mins Backup True Main Router 10.1.1.2 10.1.1.100
110 124 202 2 mins Backup True Backup 110.1.1.1 110.1.1.110

Aruba 3810M/5800R | Multicasting and Routing Guide
show vrrp IPv6 summary

Syntax
show vrrp ipv6 summary

Description
Shows useful information for all VRRP IPv6 instances on the device.

Example

```
switch# show vrrp ipv6 summary
VRRP IPv6 Summarized Information

Vlan : 10                            VRID : 123
Pri : 202                            Up Time : 5 mins
Owner : Backup                        Pre : True
Main Router Addr : fe80::5265:f3ff:feb4:29c0 State : Main Router
Virtual Addr : fe80::100
              10::11

Vlan : 3501                           VRID : 1
Pri : 198                            Up Time : 19 days
Owner : Backup                        Pre : True
Main Router Addr : fe80::e207:1bff:fec2:b543 State : Backup
Virtual Addr : aacc:1::55
              fe80::1

Vlan : 3502                           VRID : 2
Pri : 198                            Up Time : 19 days
Owner : Backup                        Pre : True
Main Router Addr : fe80::e207:1bff:fec2:b543 State : Backup
Virtual Addr : aacc:2::55
              fe80::2

Vlan : 3503                           VRID : 3
Pri : 198                            Up Time : 19 days
Owner : Backup                        Pre : True
Main Router Addr : fe80::e207:1bff:fec2:b543 State : Backup
Virtual Addr : aacc:3::55
              fe80::3
```

Error messages—Track interface

<table>
<thead>
<tr>
<th>Message</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Invalid input: out of range value</td>
<td>You have to assign a valid port or trunk to the VR instance.</td>
</tr>
<tr>
<td>Can't track a port that is part of a trunk</td>
<td>You cannot configure tracking on a port that is a member of a trunk.</td>
</tr>
<tr>
<td>Tracking is disabled on owner</td>
<td>You cannot configure a track interface on an owner VR.</td>
</tr>
<tr>
<td>Message</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>-------------</td>
</tr>
<tr>
<td>Cannot remove trunk being tracked by VRRP</td>
<td>You cannot remove a trunk that is being tracked by a VR.</td>
</tr>
<tr>
<td>Cannot enable LACP on a VRRP tracked port</td>
<td>You cannot enable LACP on a port that is being tracked by a VR.</td>
</tr>
<tr>
<td>Too many entities to track</td>
<td>You have selected too many entities to be tracked by the VR.</td>
</tr>
<tr>
<td>Cannot track trunk/LACP member</td>
<td>You cannot track the specified trunk or LACP member.</td>
</tr>
<tr>
<td>VRRP tracked port is not allowed in trunk</td>
<td>You cannot add this tracked port to a trunk.</td>
</tr>
<tr>
<td>VRRP tracked port is not allowed in LACP</td>
<td>You cannot use LACP with the tracked port.</td>
</tr>
<tr>
<td>Operation is not permitted on VR when it is configured as owner or is uninitialized.</td>
<td>The VR must be a backup and initialized in order to execute the operation.</td>
</tr>
</tbody>
</table>

**Border Gateway Protocol (BGP)**

**Introduction**

BGPv4 (RFC 4271) is the defacto internet exterior gateway protocol used between ISPs. The characteristics of BGP are:

- Controls route propagation and the selection of optimal routes, rather than route discovery and calculation, which makes BGP different from interior gateway protocols such as OSPF and RIP.
- Uses TCP to enhance reliability.
- Supports CIDR.
- Reduces bandwidth consumption by advertising only incremental updates, which allows advertising large amounts of routing information on the Internet.
- Eliminates routing loops completely by adding AS path information to BGP routes.
- Provides policies to implement flexible route filtering and selection.
- Provides scalability.

A router that advertises BGP messages is called a BGP speaker. The BGP speaker establishes peer relationships with other BGP speakers to exchange routing information. When a BGP speaker receives a new route or a route better than the current one from another AS, it advertises the route to all the other BGP peers in the local AS.

BGP can be configured to run on a router in the following two modes:

- iBGP (internal BGP)
- eBGP (external BGP)

When a BGP speaker peers with another BGP speaker that resides in the same autonomous system, the session is referred to as an iBGP session. When a BGP speaker peers with a BGP speaker that resides in a different autonomous system, the session is referred to as an eBGP session.

**Configuring BGP globally**
<table>
<thead>
<tr>
<th>Command syntax</th>
<th>Description</th>
<th>Default</th>
<th>CLI reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>router bgp as-#</td>
<td>Configures a BGP routing process.</td>
<td>Not enabled.</td>
<td>Configuring a BGP routing process on page 320</td>
</tr>
<tr>
<td>no router bgp</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>bgp router-id router-id</td>
<td>Configures a fixed router ID for the local Border Gateway Protocol (BGP) routing process.</td>
<td></td>
<td>Configuring a fixed router ID for local BGP routing process on page 320</td>
</tr>
<tr>
<td>no bgp router id</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Parameters network ipv4/mask</td>
<td>To specify the networks to be advertised by the Border Gateway Protocol (BGP) routing processes, use the network command.</td>
<td></td>
<td>Specifying the networks to be advertised by the BGP routing process on page 320</td>
</tr>
<tr>
<td>[route-map route-map-name]</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Parameters bgp timers keep-alive hold-time</td>
<td>To adjust BGP network timers, use the bgp timers command in router configuration mode.</td>
<td></td>
<td>Adjusting BGP network timers on page 320</td>
</tr>
<tr>
<td>disable</td>
<td>Re-enables the state contained within this node and all child nodes of the Border Gateway Protocol (BGP) process.</td>
<td>Disabled</td>
<td>Re-enabling state contained within nodes of BGP processes on page 320</td>
</tr>
</tbody>
</table>
Configuring a BGP routing process

Syntax:
```
router bgp as#
```
```
no router bgp
```

Configures a BGP routing process. To remove the routing process, use the `no` form of the command. This command is used in the configuration context only. This command allows you to set up a distributed routing core that automatically guarantees the loop-free exchange of routing information between autonomous systems.

Configuring a fixed router ID for local BGP routing process

Syntax:
```
bgp router-id router-id
no bgp router id
```

Configures a fixed router ID for the local Border Gateway Protocol (BGP) routing process. To remove the fixed router ID from the running configuration file and restore the default router ID selection, use the `no` form of this command.

The `bgp router-id` command is used to configure a fixed router ID for a local BGP routing process. The router ID is entered in the IP address format. Any valid IP address can be used.

Specifying the networks to be advertised by the BGP routing process

Syntax:
```
no network ipv4/mask [route-map route-map-name]
```

To specify the networks to be advertised by the Border Gateway Protocol (BGP) routing processes, use the `network` command. To remove an entry from the routing table, use the `no` form of this command.

BGP networks can be learned from connected routes, from dynamic routing, and from static route sources. The maximum number of network commands you can use is determined by the resources of the router, such as the configured NVRAM or RAM.

Adjusting BGP network timers

Syntax:
```
Parameters bgp timers keep-alive hold-time
```

To adjust BGP network timers, use the `bgp timers` command in router configuration mode. To reset the BGP timing defaults, use the `no` form of this command.

Re-enabling state contained within nodes of BGP processes

Syntax:
```
Parameters enable
disable
```

Re-enables the state contained within this node and all child nodes of the Border Gateway Protocol (BGP) process. The `disable` command disables the state contained within this node and all child nodes. The default is for the state to be disabled.

Configuring BGP policy globally
<table>
<thead>
<tr>
<th>Command syntax</th>
<th>Description</th>
<th>Default</th>
<th>CLI reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parameters bgp open-on-accept</td>
<td>Delays sending the BGP Open message until an OPEN message is received.</td>
<td></td>
<td>Delays sending the BGP open message on page 323</td>
</tr>
<tr>
<td>Parameters bgp maximum-prefix <strong>max-routes</strong></td>
<td>Specifies the maximum number of routes that BGP will accept for installation into the Routing Information Base (RIB).</td>
<td></td>
<td>Specifying the maximum routes that BGP will accept for installation into the RIB on page 323</td>
</tr>
<tr>
<td>Parameters bgp always-compare-med</td>
<td>Enables the comparison of the Multi Exit Discriminator (MED) for paths from neighbors in different autonomous systems.</td>
<td></td>
<td>Enabling comparison of MED for paths from neighbors in different autonomous systems on page 324</td>
</tr>
<tr>
<td>Parameters bgp allowas-in <strong>num-loops</strong></td>
<td>Specifies the number of time an Autonomous System number can appear in the AS <em>PATH</em>.</td>
<td></td>
<td>Enabling comparison of MED for paths from neighbors in different autonomous systems on page 324</td>
</tr>
<tr>
<td>Parameters bgp bestpath as-path-ignore</td>
<td>Configures Border Gateway Protocol (BGP) to not consider the autonomous system (AS)-path during best path route selection.</td>
<td>By default, the AS-path is considered during BGP best path selection.</td>
<td>Configuring BGP to not consider AS_PATH on page 324</td>
</tr>
<tr>
<td>Parameters bgp bestpath compare-originator-id</td>
<td>Specifies to break</td>
<td></td>
<td>Breaking ties</td>
</tr>
<tr>
<td>Command syntax</td>
<td>Description</td>
<td>Default</td>
<td>CLI reference</td>
</tr>
<tr>
<td>----------------------------------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>---------</td>
<td>--------------------------------</td>
</tr>
<tr>
<td><strong>Parameters bgp bestpath compare-router-id</strong></td>
<td>To configure a Border Gateway Protocol (BGP) routing process to compare identical routes received from different external peers during the best path selection process and to select the route with the lowest router ID as the best path, use the <code>bgp bestpath compare-router-id</code> command in router configuration mode.</td>
<td></td>
<td>Comparing identical routes received from different external peers on page 324</td>
</tr>
<tr>
<td><strong>Parameters bgp bestpath med-missing-as-worst</strong></td>
<td>To configure a Border Gateway Protocol (BGP) routing process to assign a value of infinity (max possible) to routes that are missing the Multi Exit Discriminator (MED) attribute (making the path without a MED value the least desirable path), use the <code>bgp bestpath med missing-as-worst</code> command in</td>
<td></td>
<td>Assigning value of infinity to routes missing MED attribute on page 324</td>
</tr>
<tr>
<td>Command syntax</td>
<td>Description</td>
<td>Default</td>
<td>CLI reference</td>
</tr>
<tr>
<td>----------------</td>
<td>-------------</td>
<td>---------</td>
<td>---------------</td>
</tr>
<tr>
<td>Parameters bgp default-metric <strong>med-out</strong></td>
<td>Causes a BGP MED to be set on routes when they are advertised to peers.</td>
<td></td>
<td>Setting BGP MED on routes when advertised to peers on page 325</td>
</tr>
<tr>
<td>Parameters distance bgp <strong>ext-dist</strong> int-dist loc-dist</td>
<td>A route’s preference specifies how active routes that are learned from BGP (compared to other protocols) will be selected.</td>
<td></td>
<td>Specifying a route’s preference on page 325</td>
</tr>
<tr>
<td>Parameters bgp client-to-client-reflection</td>
<td>Enables or disables client-to-client route reflection.</td>
<td>When acting as a route-reflector, this functionality is enabled by default.</td>
<td>Enabling client-to-client route reflection on page 325</td>
</tr>
<tr>
<td>Parameters bgp cluster-id <strong>ip-address</strong></td>
<td>Specifies the cluster ID to be used when the BGP router is used as a route-reflector.</td>
<td>The cluster ID default is the router ID.</td>
<td>Specifying cluster ID when BGP router is route-reflector on page 325</td>
</tr>
</tbody>
</table>

**Delaying sending the BGP open message**

**Syntax:**

Parameters bgp open-on-accept

Delays sending the BGP Open message until an OPEN message is received. When this command is specified, an OPEN message will be immediately sent when the TCP connection has completed for configured peers. If the peer is not configured (is matched by an allow clause, but not a peer command), it will continue to wait for the OPEN message from the remote peer before sending its own BGP OPEN message.

**Specifying the maximum routes that BGP will accept for installation into the RIB**

**Syntax:**

Parameters bgp maximum-prefix **max-routes**
Enabling comparison of MED for paths from neighbors in different autonomous systems

Syntax:
Parameters bgp always-compare-med

Enables the comparison of the Multi Exit Discriminator (MED) for paths from neighbors in different autonomous systems. To disallow the comparison, use the no form of this command.

The MED is one of the parameters that is considered when selecting the best path among many alternative paths. The path with a lower MED is preferred over a path with a higher MED. During the best-path selection process, MED comparison is done only among paths from the same autonomous system. The bgp always-compare-med command is used to change this behavior by enforcing MED comparison between all paths, regardless of the autonomous system from which the paths are received.

Specifying the number of times an AS number can appear in AS_PATH

Syntax:
Parameters bgp allowas-in num-loops

Specifies the number of times an Autonomous System number can appear in the AS_PATH. Use the no form of the command to set the parameter to its default value of ‘1’.

Configuring BGP to not consider AS_PATH

Syntax:
Parameters bgp bestpath as-path-ignore

Configures Border Gateway Protocol (BGP) to not consider the autonomous system (AS)-path during best path route selection. To restore default behavior and configure BGP to consider the AS-path during route selection, use the no form of this command. By default, the AS-path is considered during BGP best path selection.

Breaking ties between routes based on originator ID value

Syntax:
Parameters bgp bestpath compare-originator-id

Specifies to break ties between routes based the Originator ID value instead of the neighbor’s router ID. Use the no form of the command to not compare routes based on originator ID.

Comparing identical routes received from different external peers

Syntax:
Parameters bgp bestpath compare-router-id

To configure a Border Gateway Protocol (BGP) routing process to compare identical routes received from different external peers during the best path selection process and to select the route with the lowest router ID as the best path, use the bgp bestpath compare-router-id command in router configuration mode. To return the BGP routing process to the default operation, use the no form of this command.

The behavior of this command is disabled by default; BGP selects the route that was received first when two routes with identical attributes are received.

Assigning value of infinity to routes missing MED attribute

Syntax:
Parameters bgp bestpath med-missing-as-worst
To configure a Border Gateway Protocol (BGP) routing process to assign a value of infinity (max possible) to routes that are missing the Multi Exit Discriminator (MED) attribute (making the path without a MED value the least desirable path), use the `bgp bestpath med missing-as-worst` command in router configuration mode. To return the router to the default behavior (assign a value of 0 to the missing MED), use the no form of this command.

### Setting BGP MED on routes when advertised to peers

**Syntax:**
```
bgp default-metric med-out
```

Causes a BGP MED to be set on routes when they are advertised to peers. This value applies to all BGP peers. It can be overridden on a per-peer basis. The no form of this command, `no default-metric`, removes the configured value.

### Specifying a route's preference

**Syntax:**
```
parameters distance bgp ext-dist int-dist loc-dist
```

A route’s preference specifies how active routes that are learned from BGP (compared to other protocols) will be selected. When a route has been learned from more than one protocol, the active route will be selected from the protocol with the lowest preference. Each protocol has a default preference in this selection. This preference can be overridden by a preference value specified on the peer.

### Enabling client-to-client route reflection

**Syntax:**
```
bgp client-to-client-reflection
```

Enables or disables client-to-client route reflection. When acting as a route-reflector, this functionality is enabled by default.

### Specifying cluster ID when BGP router is route-reflector

**Syntax:**
```
bgp cluster-id ip-address
```

Specifies the cluster ID to be used when the BGP router is used as a route-reflector. The cluster ID default is the router ID.

### BGP graceful restart

**Table 39: Graceful restart commands**

<table>
<thead>
<tr>
<th>Command syntax</th>
<th>Description</th>
<th>Default</th>
<th>CLI reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>`bgp graceful-restart {restart-time val</td>
<td>[stalepath-time val] }`</td>
<td>Configures BGP graceful restart timers. s</td>
<td></td>
</tr>
</tbody>
</table>
Command syntax | Description | Default | CLI reference
--- | --- | --- | ---
**Parameters bgp log-neighbor-changes [prefix-list prefix-list-name]** | Enables or disables BGP event logging. | | 
**Parameters neighbor ipv4-addr description desc** | Describes a neighbor. | | 

### Configuring BGP graceful restart timers

**Syntax:**

```
bgp graceful-restart {restart-time val | [stalepath-time val]}
```

Configures BGP graceful restart timers as follows:

- **restart-time**
  - The time in seconds to wait for a graceful restart capable neighbor to re-establish BGP peering.

- **stalepath-time**
  - The time in seconds to hold stale routes for a restarting peer.

### Enabling event logging

**Syntax:**

```
Parameters bgp log-neighbor-changes [prefix-list prefix-list-name]
```

Enables or disables BGP event logging. Optionally, specify a prefix-list to filter log messages from specific BGP neighbors only.

### Describing a neighbor

**Syntax:**

```
Parameters neighbor ipv4-addr description desc
```

Describes a neighbor.

### Neighbor configuration and neighbor policy configuration

**Table 40: Neighbor configuration and neighbor policy configuration commands**

<table>
<thead>
<tr>
<th>Command syntax</th>
<th>Description</th>
<th>Default</th>
<th>CLI reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>neighbor ipv4-addr remote-as as-#</td>
<td>Adds an</td>
<td></td>
<td>Adding an</td>
</tr>
<tr>
<td>Command syntax</td>
<td>Description</td>
<td>Default</td>
<td>CLI reference</td>
</tr>
<tr>
<td>----------------</td>
<td>-------------</td>
<td>---------</td>
<td>---------------</td>
</tr>
<tr>
<td></td>
<td>entry to the BGP neighbor table in router configuration mode.</td>
<td></td>
<td>entry to the BGP neighbor table in router configuration mode on page 331</td>
</tr>
<tr>
<td>Parameters neighbor ipv4-addr dynamic</td>
<td>Specifies whether to enable or disable dynamic capabilities.</td>
<td></td>
<td>Enabling or disabling dynamic capabilities on page 331</td>
</tr>
<tr>
<td>Parameters neighbor ipv4-addr updated-source ipv4-addr</td>
<td>Specifies the IP address to be used on the local end of the TCP connection with the peer.</td>
<td></td>
<td>Specifying the IP address for local end of TCP connection with peer on page 332</td>
</tr>
<tr>
<td>Parameters neighbor ipv4-addr allowas-in num-loops</td>
<td>Specifies the number of times this autonomous system can appear in an AS path.</td>
<td></td>
<td>Specifying the number of times the autonomous system can appear in an AS path on page 332</td>
</tr>
<tr>
<td>Parameters neighbor ipv4-addr as-override</td>
<td>Causes all occurrences of our peer’s AS to be replaced with one from an export.</td>
<td></td>
<td>Replacing occurrences of peer's AS with one from an export on page 332</td>
</tr>
<tr>
<td>Command syntax</td>
<td>Description</td>
<td>Default</td>
<td>CLI reference</td>
</tr>
<tr>
<td>----------------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>-------------------------------------------------------------------------</td>
<td>--------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Parameters neighbor ipv4-addr ignore-leading-as</td>
<td>Some routers are capable of propagating routes without appending their own autonomous system number to the AS Path.</td>
<td>By default, BGP will drop such routes.</td>
<td>Allowing BGP to keep routes without AS number on page 332</td>
</tr>
<tr>
<td>Parameters neighbor ipv4-addr local-as as-#</td>
<td>Identifies the autonomous system (AS) that BGP is representing to a peer.</td>
<td>The default AS number for this command is the current AS (configured with the router bgp command in Global Configuration mode.)</td>
<td>Identifying the AS that BGP is representing to a peer on page 332</td>
</tr>
<tr>
<td>Parameters neighbor ipv4-addr maximum-prefix max-routes</td>
<td>Specifies the maximum number of routes that BGP will accept for installation into the RIB.</td>
<td>The value defaults to &quot;unlimited&quot; if not specified, or if using the no version of the command.</td>
<td>Specifying maximum number of routes for installation into the RIB on page 333</td>
</tr>
<tr>
<td>Parameters neighbor ipv4-addr out-delay sec</td>
<td>The specified integer represents the amount of time a route must be present in the routing database</td>
<td>Defaults to 0 if no specified or if un-configured by using no version of command.</td>
<td>Specifying the amount of time route is present in database before exported to BGP on page 333</td>
</tr>
<tr>
<td>Command syntax</td>
<td>Description</td>
<td>Default</td>
<td>CLI reference</td>
</tr>
<tr>
<td>------------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>-----------------------------------------------------------------------------------------------</td>
<td>---------------------------------------------------</td>
</tr>
<tr>
<td>Parameters <code>neighbor ipv4-addr weight weight</code></td>
<td>Preferences are the first criteria of comparison for route selection.</td>
<td>This value defaults to the globally configured preference if it is not specified.</td>
<td>Specifying a route’s preference on page 325</td>
</tr>
<tr>
<td>Parameters <code>neighbor ipv4-addr send-community</code></td>
<td>To specify that a community’s attribute should be sent to a BGP neighbor, use the neighbor send-community command in address family or router configuration mode.</td>
<td></td>
<td>Sending a community’s attribute to a BGP neighbor on page 333</td>
</tr>
<tr>
<td>Parameters <code>neighbor ipv4-addr use-med</code></td>
<td>Processes sending of MEDs and for handling received MEDs.</td>
<td>By default MEDs are used to choose which route to use.</td>
<td>Processing sent and received MEDs on page 333</td>
</tr>
<tr>
<td>Parameters <code>neighbor ipv4-addr timers keep-alive hold-time</code></td>
<td>To set the timers for a specific BGP peer, use the neighbor timers command in router configuration</td>
<td>The values of keep-alive and hold-time default to 60 and 180 seconds, respectively.</td>
<td>Setting the timer for a BGP peer on page 333</td>
</tr>
<tr>
<td>Command syntax</td>
<td>Description</td>
<td>Default</td>
<td>CLI reference</td>
</tr>
<tr>
<td>----------------</td>
<td>-------------</td>
<td>---------</td>
<td>---------------</td>
</tr>
<tr>
<td>clear ip bgp [neighbor ipv4-addr] [soft]</td>
<td>Resets BGP peering sessions, sends route refresh requests if 'soft'.</td>
<td></td>
<td>Resetting BGP peering sessions on page 334</td>
</tr>
<tr>
<td>Parameters neighbor ipv4-addr ibgp-multihop [ttl]</td>
<td>Enables or disables multi-hop peering with the specified EBGP peer, and optionally indicates the maximum number of hops (TTL.)</td>
<td></td>
<td>Enabling or disabling multi-hop peering on page 334</td>
</tr>
<tr>
<td>Parameters neighbor ipv4-addr next-hop-self</td>
<td>Forces BGP to use the router's outbound interface address as the next hop for the route updates to the peer.</td>
<td></td>
<td>Using the router's outbound interface address as next hop on page 334</td>
</tr>
<tr>
<td>Parameters neighbor ipv4-addr passive</td>
<td>If enabled, does not initiate a peering connection to the peer.</td>
<td></td>
<td>Specifying no peering connection to peer on page 334</td>
</tr>
<tr>
<td>Parameters neighbor ipv4-addr remove-private-as</td>
<td>Specifies whether the private AS # should be removed</td>
<td></td>
<td>Removing the private AS number from updates to EBGP peer</td>
</tr>
<tr>
<td>Command syntax</td>
<td>Description</td>
<td>Default</td>
<td>CLI reference</td>
</tr>
<tr>
<td>----------------</td>
<td>-------------</td>
<td>---------</td>
<td>---------------</td>
</tr>
<tr>
<td></td>
<td>from the as-path attribute of updates to the EBGP peer.</td>
<td></td>
<td>on page 334</td>
</tr>
<tr>
<td>Parameters neighbor ipv4-addr route-reflector-client</td>
<td>Acts as a route-reflector for the peer.</td>
<td></td>
<td>Acting as a route-reflector for the peer on page 334</td>
</tr>
<tr>
<td>Parameters neighbor ipv4-addr shutdown</td>
<td>Shuts down the BGP peering session without removing the associated peer configuration.</td>
<td></td>
<td>Shutting down the BGP peering session without removing peer configuration on page 334</td>
</tr>
<tr>
<td>Parameters neighbor ipv4-addr route-refresh</td>
<td>Enables or disables the advertisement of route-refresh capability in the Open message sent to the peer.</td>
<td></td>
<td>Enabling or disabling advertisement of route-refresh capability in open message on page 334</td>
</tr>
</tbody>
</table>

### Adding an entry to the BGP neighbor table in router configuration mode

**Syntax:**

```
neighbor ipv4-addr remote-as as-
no neighbor ipv4-addr
```

Adds an entry to the BGP neighbor table in router configuration mode. To remove an entry from the table, use the `no` form of this command.

Specifying a neighbor with an autonomous system number that matches the autonomous system number specified in the `router bgp global configuration` command identifies the neighbor as internal to the local autonomous system. Otherwise, the neighbor is considered external.

### Enabling or disabling dynamic capabilities
Syntax:
Parameters neighbor ipv4-addr dynamic

Specifies whether to enable or disable dynamic capabilities.
BGP Dynamic Capabilities allow the communication of a change in a BGP peer's capabilities without having to restart the peering session. The BGP implementation is done on a per-peer basis and in such a way that dynamic capabilities are supported as long as the BGP peer supports BGP Dynamic Capabilities. BGP advertises Dynamic Capabilities in the OPEN message. If a BGP peer advertises support for BGP Dynamic Capabilities in the OPEN message, then it turns on Dynamic Capabilities. Otherwise, the dynamic capabilities for this peer will be disabled. BGP supports the following BGP Dynamic Capabilities:

- Graceful restart
- Route refresh

Specifying the IP address for local end of TCP connection with peer

Syntax:
Parameters neighbor ipv4-addr updated-source ipv4-addr

Specifies the IP address to be used on the local end of the TCP connection with the peer. This is the address of a broadcast, NBMA or loopback interface and the local address of a point-to-point interface. For external peers, the local address must be on an interface that is shared with the peer or with the peer's gateway when a gateway is used. A session with an external peer will be opened only when an interface with the appropriate local address (through which the peer or gateway address is directly reachable) is operating. For internal peers, a peer session will be maintained when any interface with the specified local address is operating. In any case, an incoming connection will be recognized as a match for a configured peer only if it is addressed to the configured local address.

Specifying the number of times the autonomous system can appear in an AS path

Syntax:
Parameters neighbor ipv4-addr allowas-in num-loops

Specifies the number of times this autonomous system can appear in an AS path. When not configured, or when using the no version of the command, the value of as-loops is set to its default value of 1.

Replacing occurrences of peer's AS with one from an export

Syntax:
Parameters neighbor ipv4-addr as-override

Causes all occurrences of our peer's AS to be replaced with one from an export.

Allowing BGP to keep routes without AS number

Syntax:
Parameters neighbor ipv4-addr ignore-leading-as

Some routers are capable of propagating routes without appending their own autonomous system number to the AS Path. By default, BGP will drop such routes. Turning this parameter "on" allows BGP to keep these routes. This option should be used only if there is no doubt that these peers are not normal routers.

Identifying the AS that BGP is representing to a peer

Syntax:
Parameters neighbor ipv4-addr local-as as-#
Identifies the autonomous system (AS) that BGP is representing to a peer. The default AS number for this command is the current AS (configured with the `router bgp` command in Global Configuration mode.) This command is valid only for external peers.

**Specifying maximum number of routes for installation into the RIB**

**Syntax:**
```
Parameters neighbor ipv4-addr maximum-prefix max-routes
```
Specifies the maximum number of routes that BGP will accept for installation into the RIB. The value defaults to "unlimited" if not specified, or if using the `no` version of the command.

**Specifying the amount of time route is present in database before exported to BGP**

**Syntax:**
```
Parameters neighbor ipv4-addr out-delay sec
```
The specified integer represents the amount of time a route must be present in the routing database before it is exported into BGP. Defaults to 0 if not specified or if unconfigured by using the `no` version of command.

**Comparing preferences for route selection**

**Syntax:**
```
Parameters neighbor ipv4-addr weight weight
```
Preferences are the first criteria of comparison for route selection. This value defaults to the globally configured preference if it is not specified.

**Sending a community's attribute to a BGP neighbor**

**Syntax:**
```
Parameters neighbor ipv4-addr send-community
```
To specify that a community's attribute should be sent to a BGP neighbor, use the `neighbor send-community` command in address family or router configuration mode. To remove the entry, use the `no` form of this command. By default the communities attribute is sent to all peers.

**Processing sent and received MEDs**

**Syntax:**
```
Parameters neighbor ipv4-addr use-med
```
Processes sending of MEDS and handles received MEDs. When two routes to the same destination are received from different peers within the same peer as, they may have different MEDs. When choosing between these routes, assuming that nothing else makes one preferable to the other (such as configured policy), the values of the differing MEDs are used to choose which route to use. In this comparison, the route with the lowest MED is preferred. Routes without MEDs are treated as having a MED value of zero. By default, MEDs are used to choose which route to use.

**Setting the timer for a BGP peer**

**Syntax:**
```
Parameters neighbor ipv4-addr timers keep-alive hold-time
```
To set the timers for a specific BGP peer, use the `neighbor timers` command in router configuration mode. To clear the timers for a specific BGP peer, use the `no` form of this command. The values of keep-alive and hold-time default to 60 and 180 seconds, respectively.
The timers configured for a specific neighbor override the timers configured for all BGP neighbors using the timers command.

**Resetting BGP peering sessions**

**Syntax:**
clear ip bgp [neighbor ipv4-addr] [soft]
Resets BGP peering sessions; sends route refresh requests if 'soft'.

**Enabling or disabling multi-hop peering**

**Syntax:**
Parameters neighbor ipv4-addr ibgp-multihop [ttl]
Enables or disables multi-hop peering with the specified EBGP peer, and optionally indicates the maximum number of hops (TTL.)

**Using the router's outbound interface address as next hop**

**Syntax:**
Parameters neighbor ipv4-addr next-hop-self
Forces BGP to use the router's outbound interface address as the next hop for the route updates to the peer.

**Specifying no peering connection to peer**

**Syntax:**
Parameters neighbor ipv4-addr passive
If enabled, does not initiate a peering connection to the peer.

**Removing the private AS number from updates to EBGP peer**

**Syntax:**
Parameters neighbor ipv4-addr remove-private-as
Specifies whether the private AS # should be removed from the as-path attribute of updates to the EBGP peer.

**Acting as a route-reflector for the peer**

**Syntax:**
Parameters neighbor ipv4-addr route-reflector-client
Acts as a route-reflector for the peer.

**Shutting down the BGP peering session without removing peer configuration**

**Syntax:**
Parameters neighbor ipv4-addr shutdown
Shuts down the BGP peering session without removing the associated peer configuration.

**Enabling or disabling advertisement of route-refresh capability in open message**

**Syntax:**
Parameters neighbor ipv4-addr route-refresh
Enables or disables the advertisement of route-refresh capability in the Open message sent to the peer.
Synchronizing BGP-IGP

**Table 41: BGP-IGP synchronization commands**

<table>
<thead>
<tr>
<th>Command syntax</th>
<th>Description</th>
<th>CLI reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parameters redistribute protocol [route-map route-map-name]</td>
<td>Specifies routes to export into BGP. This command causes routes from the specified protocol to be considered for redistribution into BGP.</td>
<td>Specifying routes to export into BGP on page 335</td>
</tr>
<tr>
<td>Parameters neighbor ipv4-addr route-map route-map-name [[in]</td>
<td>[out]]</td>
<td>Route maps control the redistribution of routes between protocols.</td>
</tr>
</tbody>
</table>

**Specifying routes to export into BGP**

**Syntax:**

Parameters redistribute protocol [route-map route-map-name]

Specifies routes to export into BGP. This command causes routes from the specified protocol to be considered for redistribution into BGP. Additionally, if a route map is specified, then routes from the specified protocol that match the named route map will be considered for redistribution into the current protocol. If the referenced route map has not yet been configured, then an empty route map is created with the specified name.

**Specifying route map to be exported in or out of BGP**

**Syntax:**

Parameters neighbor ipv4-addr route-map route-map-name [[in] | [out]]

Route maps control the redistribution of routes between protocols. Only after configuring a route map, can it then be specified in BGP. Use this command to specify a configured route map to be exported into or out of BGP. When the in version of this command is configured, all IPv4 announcements received from the specified neighbor should be run against the policy specified in the named route-map. When the out version of this command is used, it specifies that all IPv4 announcements sent to the specified neighbor should be run against the policy.
specified in the named route-map. After evaluating this policy, each route will be compared to the specified route-target export, to see if announcement is acceptable.

**BGP path attributes**

**Classification of path attributes**

There are four categories of path attributes:

- **Well-known mandatory**
  Must be recognized by all BGP routers and must be included in every update message. Routing information errors occur without this attribute.

- **Well-known discretionary**
  Can be recognized by all BGP routers; can be included in every update message as needed.

- **Optional transitive**
  Transitive attribute between ASs. A BGP router not supporting this attribute can still receive routes with this attribute and advertise them to other peers.

- **Optional non-transitive**
  If a BGP router does not support this attribute, it will not advertise routes with this attribute.

The category of each BGP path attribute is described in the following table.

**Table 42: BGP path attributes**

<table>
<thead>
<tr>
<th>Name</th>
<th>Category</th>
</tr>
</thead>
<tbody>
<tr>
<td>ORIGIN</td>
<td>Well-known mandatory</td>
</tr>
<tr>
<td>AS_PATH</td>
<td>Well-known mandatory</td>
</tr>
<tr>
<td>NEXT_HOP</td>
<td>Well-known mandatory</td>
</tr>
<tr>
<td>LOCAL_PREF</td>
<td>Well-known discretionary</td>
</tr>
<tr>
<td>ATOMIC_AGGREGATE</td>
<td>Well-known discretionary</td>
</tr>
<tr>
<td>COMMUNITY</td>
<td>Optional transitive</td>
</tr>
<tr>
<td>MULTI_EXIT_DISC (MED)</td>
<td>Optional non-transitive</td>
</tr>
<tr>
<td>ORIGINATOR_ID</td>
<td>Optional non-transitive</td>
</tr>
<tr>
<td>CLUSTER_LIST</td>
<td>Optional non-transitive</td>
</tr>
</tbody>
</table>

**Using BGP path attributes**

**ORIGIN**

ORIGIN is a well-known mandatory attribute that defines the origin of routing information, that is, how a route became a BGP route. There are three types:

- IGP
Has the highest priority. Routes added to the BGP routing table using the network command have the IGP attribute.

**EGP**

Has the second highest priority. Routes obtained via EGP have the EGP attribute.

**Incomplete**

Has the lowest priority. The source of routes with this attribute is unknown, which does not mean such routes are unreachable. The routes that are redistributed from other routing protocols have this attribute.

**AS_PATH**

AS_PATH is a well-known mandatory attribute. This attribute identifies the autonomous systems through which routing information carried in the Update message has passed. When a route is advertised from the local AS to another AS, each passed AS number is added into the AS_PATH attribute, allowing the receiver to determine the ASs for routing back the message. The number of the AS closest to the receiver's AS is leftmost, as shown in the following figure.

**Figure 13  AS_PATH attribute**

![AS_PATH Attribute Diagram](image)

Usually a BGP router does not receive routes containing the local AS number to avoid routing loops.

---

**NOTE**

The current implementation supports using the `neighbor allow-as-loop` command to receive routes containing the local AS number.

---

The AS_PATH attribute can be used for route selection and filtering. BGP gives priority to the route with the shortest AS_PATH length if other factors are the same. As shown in the above figure, the BGP router in AS50 gives priority to the route passing AS40 for sending data to the destination 8.0.0.0.

In some applications, you can apply a routing policy to control BGP route selection by modifying the AS_PATH length.

By configuring an AS path filtering list, you can filter routes based on AS numbers contained in the AS_PATH attribute.

**NEXT_HOP**

Different from IGP, the NEXT_HOP attribute may not be the IP address of a directly connected router. It involves three types of values, as shown in the following figure.
- When advertising a self-originated route to an eBGP peer, a BGP speaker sets the NEXT_HOP for the route to the address of its sending interface.
- When sending a received route to an eBGP peer, a BGP speaker sets the NEXT_HOP for the route to the address of the sending interface.
- When sending a route received from an eBGP peer to an iBGP peer, a BGP speaker does not modify the NEXT_HOP attribute. If load-balancing is configured, the NEXT_HOP attribute will be modified. For load-balancing information, refer to BGP route selection.

**Figure 14** NEXT_HOP attribute

![Diagram of AS 100, AS 200, and AS 300 with BGP and EBGP connections]

**MED (MULTI_EXIT_DISC)**

The MED attribute is exchanged between two neighboring ASs, each of which does not advertise the attribute to any other AS. Similar to metrics used by IGP, MED is used to determine the best route for traffic going into an AS.

When a BGP router obtains multiple routes to the same destination but with different next hops, it considers the route with the smallest MED value the best route if other conditions are the same. As shown below, traffic from AS10 to AS20 travels through Router B that is selected according to MED.

**Figure 15** MED attribute

![Diagram of MED attribute with AS10, Router A, Router B, Router C, and Router D]

In general, BGP compares MEDs of routes received from the same AS only.

---

**NOTE**

The current implementation supports using the always-compare-med command to force BGP to compare MED values of routes received from different ASs.
**LOCAL_PREF**

The LOCAL_PREF attribute is exchanged between iBGP peers only, and therefore is not advertised to any other AS. It indicates the priority of a BGP router. LOCAL_PREF is used to determine the best route for traffic leaving the local AS. When a BGP router obtains from several iBGP peers multiple routes to the same destination but with different next hops, it considers the route with the highest LOCAL_PREF value as the best route. As shown below, traffic from AS20 to AS10 travels through Router C that is selected according to LOCAL_PREF.

**Figure 16** LOCAL_PREF attribute

![LOCAL_PREF Diagram](image)

**COMMUNITY**

The COMMUNITY attribute is used to simplify routing policy usage, and to ease management and maintenance. It identifies a collection of destination addresses having identical attributes, without physical boundaries in between, and having nothing to do with the local AS. Well known community attributes involve:

**Internet**

By default, all routes belong to the Internet community. Routes with this attribute can be advertised to all BGP peers.

**No_EXPORT**

After being received, routes with this attribute cannot be advertised out the local AS.

**No_Advertise**

After being received, routes with this attribute cannot be advertised to other BGP peers.

**No_EXPORT_Subconfed**

After being received, routes with this attribute cannot be advertised out the local AS.

**BGP route selection**

**Route selection rules**

The current BGP implementation supports the following route selection sequence:
- Prefer the route with the lowest Administrative Distance.
- Prefer the route with the larger weight.
- Prefer the route with the highest LOCAL_PREF value.
- Prefer the path that was locally originated via a network or through redistribution from an IGP.
- Prefer the route with the shortest path, excluding confederation segments.
- Prefer the route with the “best” ORIGIN. IGP is better than EGP, which is better than Incomplete.
- If bgp always-compare-med is not configured, prefer any routes that do not have an inferior MED. If bgp always-compare-med has been configured, prefer the route with the lowest MED.
- Prefer the route with the lowest IGP cost to the BGP next hop. IGP cost is determined by comparing the preference, then the weight, then the metric, and finally the metric of the two resolving routes.
- If “ip load-sharing” is enabled, BGP inserts up to n most recently received paths in the IP routing table. This allows eBGP multipath load sharing. The maximum value of n is currently 4. The default value of n, when “ip load-sharing” is disabled, is 1. The oldest received path is marked as the best path in the output of `show ip bgp prefix/len`.
- Prefer routes received from external peers.
- If bgp tie-break-on-age has been specified, prefer the older route.
- If
  
  ```
  bgp bestpath
  compare-router-id
  has been specified, prefer the route learned with the lowest router ID. The router ID is taken from the Open message of the peering session over which the route was received, unless bgp bestpath compare-originator-id has been specified, and the route was received with an ORIGIN_ID. In the latter case, the ORIGIN_ID is used instead of the router ID from the Open message.
  ```
- If
  
  ```
  bgp bestpath
  compare-cluster-list-length
  has been specified, prefer the route with the lowest CLUSTER_LIST length.
  ```
- Prefer the route with the lowest neighbor address.

---

**NOTE**

`CLUSTER_IDS` of route reflectors form a CLUSTER_LIST. If a route reflector receives a route that contains its own CLUSTER ID in the CLUSTER_LIST, the router discards the route to avoid routing loops.

---

**Recursive route in iBGP**

The next Hop of an iBGP route may not always be directly connected. One of the reasons is next hops in routing information exchanged between iBGP peers are not modified. In this case, the BGP router needs to find the directly connected next hop via IGP. The matching route with the direct next hop is called the recursive route. The process of finding a recursive route is route recursion.

**Route selection with BGP load sharing**

BGP differs from IGP in the implementation of load balancing in the following:

- IGP routing protocols such as RIP and OSPF compute metrics of routes, and then implement load sharing over routes with the same metric and to the same destination. The route selection criterion is metric.
- BGP has no route computation algorithm, so it cannot implement load sharing according to metrics of routes.
However, BGP has abundant route selection rules, through which it selects available routes for load sharing and adds load sharing to route selection rules.

- BGP implements load sharing only on routes that have the same WEIGHT, LOCAL_PREF, ORIGIN, AS_PATH, MED and IGP COST.
- BGP load sharing is applicable between eBGP peers and between iBGP peers.
- If multiple routes to the same destination are available, BGP selects the configured number of routes for load sharing. The maximum number of routes for load sharing is currently 4. Load sharing is enabled by default.

**Figure 17** Network diagram for BGP load sharing

In Figure 17, Router D and Router E are iBGP peers of Router C. Router A and Router B both advertise a route destined for the same destination to Router C. If load sharing is configured and the two routes have the same AS_PATH attribute, ORIGIN attribute, LOCAL_PREF and MED, Router C installs both the two routes to its route table for load sharing. After that, Router C forwards to Router D and Router E the route that has AS_PATH unchanged but has NEXT_HOP changed to Router C; other BGP transitive attributes are those of the best route.

**BGP route advertisement rules**

The current BGP implementation supports the following route advertisement rules:

- When multiple feasible routes to a destination exist, the BGP speaker advertises only the best route to its peers.
- A BGP speaker advertises only routes used by itself.
- A BGP speaker advertises routes learned from an eBGP peer to all its peers, both eBGP and iBGP.
- A BGP speaker does not advertise routes learnt from an iBGP peer to its other iBGP peers.
- A BGP speaker advertises routes learnt from iBGP to eBGP peers. Note that BGP and IGP synchronization is disabled always and those routes are advertised to eBGP peers directly.

**Protocols and standards**
BGP extensions

Route reflection

By design, IBGP peers do not advertise iBGP routes to other iBGP peers. In order for iBGP peers to learn all the routes within the autonomous system as well as all the external routes, the iBGP peers would have to be fully meshed. This means for \( n \) iBGP peers there would have to be \( n(n-1)/2 \) iBGP sessions. In a large autonomous system network configuration would become an issue.

Route Reflection is one of the alternate solutions to alleviate this problem. In the BGP network, one of the iBGP speakers is designated as the route reflector. The route reflector advertises the routes it learns to other iBGP peers.

In a route reflector configuration the other iBGP peers are classified as client peers and non-client peers.

The action taken by the route reflector (after determining the best route) depends on whether the best route was received from a client peer or a non-client peer. If the route was received from a client peer, the route reflector will reflect that route to all the client peers and non-client peers.

If the route was received from a non-client peer, then the route is advertised to all its configured clients.

Route reflection introduces two new discretionary attributes: Originator ID and Cluster List, which are used in determining the best path as defined in BGP route selection on page 339.

In an Autonomous System more than one route reflector can be configured.

BGP graceful restart (GR)

When a BGP speaker shuts down, planned or unplanned, the routes that are advertised by the speaker and reachable via the speaker now become unreachable. Upon detecting that the BGP speaker has restarted, the peers delete the routes and re-add them when the restarting router advertises them again. This results in route-flap across the BGP connectivity and impacts multiple routing domains causing transient instability in the network.

The Graceful Restart capability is supported as a ‘helper router’ on the switches. In ‘helper only’ mode the router helps the other restarting router by holding the received routes from it as stale routes and not dropping them.

1. To establish a BGP session with a peer, a BGP GR Restarter sends an OPEN message with GR capability to the peer.

2. Upon receipt of this message, the peer is aware that the sending router is capable of Graceful Restart, and sends an OPEN message with GR Capability to the GR Restarter to establish a GR session. If neither party has the GR capability, the session established between them will not be GR capable.

3. The GR session between the GR Restarter and its peer goes down when the GR Restarter restarts BGP. The GR capable peer will mark all routes associated with the GR Restarter as stale. However, during the configured GR Time, it still uses these routes for packet forwarding.
4. After the restart, the GR Restarter will reestablish a GR session with its peer and send a new GR message notifying the completion of restart. Routing information is exchanged between them for the GR Restarter to create a new routing table and forwarding table with stale routing information removed. Then the BGP routing convergence is complete.

Route refresh

When the inbound policy-filter for a peer changes, the routes advertised by the peer must be presented to the policy-filter engine to take effect. This means that all the routes that were received from a peer will have to be preserved in the router and this would raise the demand on memory and CPU resources of the router. The route refresh capability allows the router to request the peer to re-advertise the routes thereby avoiding the requirement to keep a copy of all the routes that were received from all the peers.

BGP basic configuration

Configuring a BGP connection

Since BGP runs on TCP, you must specify the IP addresses of the peers in order to establish a BGP session. The peers may not be directly connected. IP addresses of loopback interfaces can be used to improve the stability of BGP connections.

Prerequisites

The neighboring nodes must be accessible to each other at the network layer.

Creating a BGP connection

- A router ID is the unique identifier of a BGP router in an AS.
- To ensure the uniqueness of a router ID and enhance network reliability, you can specify in BGP configuration context the IP address of a local loopback interface as the router ID.
- If no router ID is specified in BGP context, the global router ID is used.
- If the global router ID is used and then it is removed, the system will select a new router ID.
- Unconfiguring the router ID in BGP context can make the system select a new router ID.

Follow these steps to create a BGP connection:

<table>
<thead>
<tr>
<th>To do...</th>
<th>Use the command...</th>
<th>Remarks</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enter global configuration context</td>
<td>configuration</td>
<td></td>
</tr>
<tr>
<td>Enter BGP context</td>
<td>router bgp as-number</td>
<td>Not enabled by default</td>
</tr>
<tr>
<td>Enable BGP</td>
<td>enable</td>
<td></td>
</tr>
<tr>
<td>Specify a BGP Router ID</td>
<td>bgp router-id ip-address</td>
<td>Optional. By default, the global router ID is used.</td>
</tr>
</tbody>
</table>
### Specifying the source interface for TCP connections

BGP uses TCP as the transport layer protocol. By default, BGP uses the output interface of the optimal router to a peer as the source interface for establishing TCP connections to the peer. If a BGP router has multiple links to a peer, when the source interface fails, BGP has to reestablish TCP connections, causing network oscillation. Therefore, it is recommended to use a loopback interface as the source interface to enhance stability of BGP connections.

Follow these steps to specify the source interface of TCP connections:

<table>
<thead>
<tr>
<th>To do...</th>
<th>Use the command...</th>
<th>Remarks</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enter global configuration context</td>
<td>configuration</td>
<td></td>
</tr>
<tr>
<td>Enter BGP context</td>
<td>bgp as-number</td>
<td></td>
</tr>
<tr>
<td>Specify the source interface for establishing TCP connections to a neighbor.</td>
<td>neighbor {ip-address} update-source {ip-address}</td>
<td>Required. By default, BGP uses the outbound interface of the best route to the BGP peer as the source interface for establishing a TCP connection to the peer.</td>
</tr>
</tbody>
</table>

### Establishing MD5 authentication for TCP connections

BGP requires TCP as the transport protocol. To enhance security, you can configure BGP to perform MD5 authentication when establishing a TCP connection. The two parties must have the same password configured to establish TCP connections. BGP MD5 authentication is not for BGP packets, but for TCP connections. If the authentication fails, no TCP connection can be established.

<table>
<thead>
<tr>
<th>To do...</th>
<th>Use the command...</th>
<th>Remarks</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enter system view</td>
<td>system-view</td>
<td></td>
</tr>
<tr>
<td>Enter BGP view</td>
<td>bgp as-number</td>
<td></td>
</tr>
</tbody>
</table>
Allowing establishment of an eBGP connection to a non-directly connected peer

In general, direct physical links should be available between eBGP peers. If not, you can use the `neighbor ip-address ebgp-multihop` command to establish a TCP connection over multiple hops between two peers.

Follow these steps to allow establishment of eBGP connection to a non-directly connected peer.

<table>
<thead>
<tr>
<th>To do...</th>
<th>Use the command...</th>
<th>Remarks</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable MD5 authentication when establishing a TCP connection to the peer/peer group</td>
<td><code>neighbor &lt;ipaddress&gt; password &lt;xxxxx&gt;</code></td>
<td>Optional. Not enabled by default.</td>
</tr>
</tbody>
</table>

Controlling route distribution, reception and advertisement

**Prerequisites**
The neighboring nodes must be accessible to each other at the network layer.

**Configuring BGP Route Redistribution**
You can redistribute IGP routes into BGP. During route redistribution, BGP can filter routing information from specific routing protocols.

<table>
<thead>
<tr>
<th>To do...</th>
<th>Use the command...</th>
<th>Remarks</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enter global configuration context</td>
<td><code>configuration</code></td>
<td></td>
</tr>
<tr>
<td>Enter BGP context</td>
<td><code>bgp as-number</code></td>
<td></td>
</tr>
<tr>
<td>Allow the establishment of eBGP connection to a non-directly connected peer</td>
<td><code>neighbor ip-address ebgp-multihop [hop-count]</code></td>
<td>Optional. hop-count is 1 by default for eBGP peers</td>
</tr>
</tbody>
</table>

The ORIGIN attribute of networks advertised into the BGP routing table with the `network` command is IGP. These networks must exist in the local IP routing table. Using a routing policy makes route control more flexible.

Configuring BGP route inbound and outbound filtering policies
Follow these steps to configure BGP route reception filtering policies:

<table>
<thead>
<tr>
<th>To do...</th>
<th>Use the command...</th>
<th>Remarks</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enter global Configuration context</td>
<td>configuration</td>
<td></td>
</tr>
<tr>
<td>Enter BGP context</td>
<td>bgp as-number</td>
<td></td>
</tr>
<tr>
<td>Apply filter policy on the inbound or the outbound for each peer</td>
<td>neighbor ip-address route-map route-map-name [in</td>
<td>out]</td>
</tr>
</tbody>
</table>

**CAUTION**
Only routes permitted by the specified filtering policies can be installed into the local BGP routing table.

**Configuring BGP route attributes**

**Prerequisites**
The neighboring nodes must be accessible to each other at the network layer.

**Configuration procedure**
You can configure BGP route attributes to influence BGP route selection.
Follow these steps to configure BGP route attributes.

<table>
<thead>
<tr>
<th>To do...</th>
<th>Use the command...</th>
<th>Remarks</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enter global configuration context</td>
<td>configuration</td>
<td></td>
</tr>
<tr>
<td>Enter BGP context</td>
<td>bgp as-number</td>
<td></td>
</tr>
<tr>
<td>Configure preferences for external, internal, local routes</td>
<td>preference {external-preference internal-preference local-preference}</td>
<td>Optional. The default preferences of external, internal, and local routes are 20, 200, and 200 respectively.</td>
</tr>
<tr>
<td>Configure weight to be assigned to received routes from a peer</td>
<td>neighbor {ip-address} weight {weight}</td>
<td>Optional</td>
</tr>
<tr>
<td>Specify the router as the next hop of routes sent to a peer</td>
<td>neighbor {ip-address} next-hop-self</td>
<td>Optional. By default, advertisements to an eBGP peer take the router as the next hop, while advertisements to an iBGP peer do not take the local router as the next hop.</td>
</tr>
<tr>
<td>Configure the AS_PATH attribute:</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
To do... | Use the command... | Remarks |
---|---|---|
Configure repeating times of local AS number in routes from a peer | neighbor {ip-address}
allow-as-in [number] | Optional. The local AS number cannot be repeated in routes from the peer. |
Specify a fake AS number for a peer | neighbor {ip-address}
local-as as-number | Optional. Not specified by default. This command is only applicable to an eBGP peer. |
Substitute local AS number for the AS number of a peer in the AS_PATH attribute | neighbor {ip-address}
as-override | Optional. The substitution is not configured by default. |
Configure BGP to not keep private AS numbers in the AS_PATH attribute of updates to a peer | neighbor {ip-address}
remove-private-as | Optional. By default, BGP updates carry private AS numbers. |

- Using a routing policy can set preferences for routes matching it. Routes not matching it use the default preferences.
- If other conditions are identical, the route with the smallest MED value is selected as the best external route.
- Using the neighbor next-hop-self command can specify the router as the next hop for routes sent to a peer. If BGP load balancing is configured, the router specifies itself as the next hop for routes sent to a peer regardless of whether the neighbor next-hop-self command is configured.
- In a “third party next hop” network, that is, a BGP router has two eBGP peers in a common broadcast subnet, the BGP router does not specify itself as the next hop for routes sent to such an eBGP peer, unless the neighbor next-hop-self command is configured.
- BGP checks if the AS_PATH attribute of a route from a peer contains the local AS number. If so, it discards the route to avoid routing loops.
- You can specify a fake AS number to hide the real one. The fake AS number applies to routes sent to eBGP peers only, that is, eBGP peers in other ASs can only find the fake AS number.
- The neighbor as-override command is used only in specific networking environments. Inappropriate use of the command may cause routing loops.

### Tuning and optimizing BGP networks

#### Prerequisites

The neighboring nodes must be accessible to each other at the network layer.

#### Configuring a BGP keepalive interval and holdtime
After establishing a BGP connection, two routers send keepalive messages periodically to each other to keep the connection. If a router receives no keepalive or update message from the peer within the holdtime, it breaks the connection.

If two parties have the same timer assigned with different values, the smaller one is used.

Follow these steps to configure BGP keepalive interval and holdtime.

<table>
<thead>
<tr>
<th>To do...</th>
<th>Use the command...</th>
<th>Remarks</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enter global configuration context</td>
<td>configuration</td>
<td></td>
</tr>
<tr>
<td>Enter BGP context</td>
<td>bgp as-number</td>
<td></td>
</tr>
<tr>
<td>Configure the global keepalive interval and holdtime</td>
<td>timers {keepalive-time} {hold-time}</td>
<td></td>
</tr>
<tr>
<td>Configure the keepalive interval and holdtime for a peer</td>
<td>neighbor {ip-address} timers {keepalive-time} {hold-time}</td>
<td>Optional. By default, the keepalive interval is 60 seconds, and holdtime is 180 seconds.</td>
</tr>
</tbody>
</table>

- The maximum keepalive interval should be one third of the holdtime and no less than 1 second. The holdtime is no less than 3 seconds unless it is set to 0.
- Intervals set with the neighbor timers command are preferred to those set with the timers command.
- If the router has established a neighbor relationship with a peer, you need to reset the BGP connection to validate the new set timers.

Configuring a large scale BGP network

In a large-scale BGP network, configuration and maintenance become difficult due to large numbers of BGP peers. To facilitate configuration in this case, you can configure community or route reflector as needed.

Prerequisites

The neighboring nodes must be accessible to each other at the network layer.

Configuring a BGP route reflector

Follow these steps to configure a BGP route reflector:

<table>
<thead>
<tr>
<th>To do...</th>
<th>Use the command...</th>
<th>Remarks</th>
</tr>
</thead>
<tbody>
<tr>
<td>Entere the global configuration context</td>
<td>configuration</td>
<td></td>
</tr>
<tr>
<td>Enter the BGP context</td>
<td>bgp as-number</td>
<td></td>
</tr>
<tr>
<td>Configure the router as a route reflector and</td>
<td>client-to-client-reflection</td>
<td>Enabled by default</td>
</tr>
</tbody>
</table>
**To do...** | **Use the command...** | **Remarks**
---|---|---
specify a peer as its client |  |  
Enable route reflection between clients | `neighbor {ip-address} route-reflector-client` | Optional. Enabled by default. 

It is not required to make clients of a route reflector fully meshed. The route reflector forwards routing information between clients. If clients are fully meshed, you can disable route reflection between clients to reduce routing costs.

**CAUTION**

A cluster has only one route reflector, and the router ID is used to identify the cluster. You can configure multiple route reflectors to improve network stability. In this case, you must specify the same cluster ID for these route reflectors to avoid routing loops.

---

**Configuring BGP graceful restart (GR)**

Perform the following configuration on the GR Restarter and GR Helper respectively.

**NOTE**

A device can act as both the GR Restarter and GR Helper simultaneously.

Follow these steps to configure BGP GR.

---

**To do...** | **Use the command...** | **Remarks**
---|---|---
Enter the global Configuration context | `configuration` |  
Enable BGP, and enter its view | `bgp as-number` |  
Configure graceful restart | `bgp graceful-restart staleparth-time {stale-path-time}` | Required. Disabled by default. 
Configure the maximum time allowed for the peer to reestablish a BGP session | `graceful-restart timer restart timer` | Optional. 120 seconds by default. 
Configure the maximum time to wait for the End-of-RIB marker | `graceful-restart timer` |  

The maximum time allowed for the peer (the GR restarter) to reestablish a BGP session should be less than the Holdtime carried in the OPEN message.

**NOTE**

The End-Of-RIB (End of Routing-Information-Base) indicates the end of route updates.

---

**Displaying information about BGP configuration**

**Displaying BGP information**
<table>
<thead>
<tr>
<th>To do...</th>
<th>Use the command...</th>
<th>Remarks</th>
</tr>
</thead>
<tbody>
<tr>
<td>Display information about BGP routes installed in the BGP routing information base (RIB)</td>
<td><code>show ip bgp</code></td>
<td>Available in any view</td>
</tr>
<tr>
<td>Display specific information about the route and the BGP path attributes of the route</td>
<td><code>show ip bgp ipv4-addr/masklen</code></td>
<td></td>
</tr>
<tr>
<td>Display generic global configuration information regarding BGP</td>
<td><code>show ip bgp general</code></td>
<td></td>
</tr>
<tr>
<td>Display detailed information on the route if the route’s AS_PATH information matches the supplied regular expression</td>
<td><code>show ip bgp ipv4-addr/masklen regexp aspath-reg-ex</code></td>
<td></td>
</tr>
<tr>
<td>Display all the routes in the IP routing table, including BGP routes</td>
<td><code>show ip route</code></td>
<td></td>
</tr>
<tr>
<td>Display only the BGP routes in the IP routing table</td>
<td><code>show ip route bgp [ip4-addr]</code></td>
<td></td>
</tr>
<tr>
<td>Display the routes whose community information matches the supplied</td>
<td><code>show ip bgp communitycomm-num... regexp aspath-reg-ex</code></td>
<td></td>
</tr>
<tr>
<td>To do...</td>
<td>Use the command...</td>
<td>Remarks</td>
</tr>
<tr>
<td>---</td>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>community numbers and also the AS_PATH information matches the supplied regular expression</td>
<td>show ip bgp communitycomm-num...exact regexp aspath-reg-ex</td>
<td></td>
</tr>
<tr>
<td>Display the routes whose community information matches exactly the supplied community numbers and also whose AS_PATH information matches the supplied regular expression</td>
<td>show ip bgp regex reg-ex</td>
<td></td>
</tr>
<tr>
<td>Display all routes whose AS_PATH matches the regular-expression given</td>
<td>show ip bgp [ipv4-addr</td>
<td>masklen [longer-prefix]] route community</td>
</tr>
<tr>
<td>Display basic route information (destination and nexthop) and the communities tagged to the route in full</td>
<td>show ip bgp neighbor [ip4-addr]</td>
<td></td>
</tr>
<tr>
<td>Display BGP peer information</td>
<td>show ip bgp summary</td>
<td></td>
</tr>
<tr>
<td>Display in brief the BGP neighbor information</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
To do... | Use the command... | Remarks
--- | --- | ---
Display the list of AS_PATH that BGP has learned from the routing information it has received | `show ip bgp as-path` |  
Display the list of protocols whose routes are being redistributed into BGP | `show ip bgp redistribute` |  

### BGP configuration examples

#### BGP basic configuration

**Network requirements**

In the following network, run eBGP between Switch A and Switch B and iBGP between Switch B and Switch C so that Switch C can access the network 8.1.1.0/24 connected to Router A.

**Figure 18  Network diagram for BGP basic configuration**

![Network diagram for BGP basic configuration](image)

**Configuration procedure**

1. Configure IP addresses for interfaces (omitted.)
2. Configure iBGP.
   a. To prevent route flapping caused by port state changes, this example uses loopback interfaces to establish iBGP connections. Because loopback interfaces are virtual interfaces, you need to use the `peer connect-interface` command to specify the loopback interface as the source interface for establishing BGP connections.
   b. Enable OSPF in AS 65009 to ensure that Switch B can communicate with Switch C through loopback interfaces.
3. Configure Switch B

```
switch(config)# router bgp 65009
switch(bgp)# bgp router-id 2.2.2.2
switch(bgp)# neighbor 3.3.3.3 remote-as 65009
switch(bgp)# exit
switch(config)# router ospf
switch(ospf)# enable
switch(ospf)# area 0
switch(ospf)# network 2.2.2.2/32
switch(ospf)# network 9.1.1.1/24
switch(ospf)# exit
switch(config)# vlan 300
switch(vlan-300)# ip ospf
```

4. Configure Switch C

```
switch(config)# router bgp 65009
switch(bgp)# bgp router-id 3.3.3.3
switch(bgp)# neighbor 2.2.2.2 remote-as 65009
switch(bgp)# neighbor 2.2.2.2 connect-interface loopback 0
switch(bgp)# exit
switch(config)# router ospf
switch(ospf)# enable
switch(ospf)# area 0
switch(ospf)# network 3.3.3.3/32
switch(ospf)# network 9.1.1.0/24
switch(ospf)# exit
switch(config)# vlan 300
switch(vlan-300)# ip ospf
switch(vlan-300)# show ip bgp summary

Peer Information

<table>
<thead>
<tr>
<th>Remote Address</th>
<th>Remote-AS</th>
<th>Local-AS</th>
<th>State</th>
<th>Admin Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.2.2.2</td>
<td>65009</td>
<td>65009</td>
<td></td>
<td>Established</td>
</tr>
</tbody>
</table>
```

5. The output information shows that Switch C has established an iBGP peer relationship with Switch B.

6. Configure eBGP.

   a. The eBGP peers, Switch A and Switch B (usually belonging to different carriers), are located in different ASs. Their loopback interfaces are not reachable to each other, so directly connected interfaces are used for establishing BGP sessions. To enable Switch C to access the network 8.1.1.0/24 that is connected directly to Switch A, add network 8.1.1.0/24 to the BGP routing table of Switch A.

7. Configure Switch A.

```
switch(config)# router bgp 65008
switch(bgp)# bgp router-id 1.1.1.1
switch(bgp)# neighbor 3.1.1.1 remote-as 65009
switch(bgp)# network 8.1.1.1/24
switch(bgp)# exit
```

8. Configure Switch B.

```
switch(config)# router bgp 65009
switch(bgp)# neighbor 3.1.1.2 remote-as 65008
switch(bgp)# exit
```
9. Show IP bgp peer information on Switch B.

```
switch(config)# show ip bgp summary

Peer Information

Remote Address  Remote-AS  Local-AS  State       Admin Status
-----------------  --------  -------  ---------  ------------
2.2.2.2           65009    65009   Established  Start
3.1.1.2           65008    65009   Established  Start
```

10. The output shows that Switch B has established an iBGP peer relationship with Switch C and an eBGP peer relationship with Switch A.

11. Display the BGP routing table on Switch A.

```
switch(bgp)# show ip bgp

Local AS : 100
Local Router-id : 20.0.0.1
BGP Table Version : 0
Status codes: * - valid, > - best, i - internal, e - external, s - stale
Origin codes: i - IGP, e - EGP, ? - incomplete

Network      Nexthop  Metric  LocalPref  Weight  AsPath
-------------  -------  ------  --------  -------  --------
*>  8.1.1.0/24  0       32768  I
*>  8.1.1.0/24  0.0.0.0  0       0   I
```

12. Display the BGP routing table on Switch B.

```
switch# show ip bgp

Local AS : 100
Local Router-id : 20.0.0.1

BGP Table Version : 0
Status codes: * - valid, > - best, i - internal, e - external, s - stale
Origin codes: i - IGP, e - EGP, ? - incomplete

Network      Nexthop  Metric  LocalPref  Weight  AsPath
-------------  -------  ------  --------  -------  --------
*>e  8.1.1.0/24  0       0       0  65008i
```

13. Display the BGP routing table on Switch C.

```
switch(bgp)# show ip bgp

Local AS : 100
Local Router-id : 20.0.0.1

BGP Table Version : 0
Status codes: * - valid, > - best, i - internal, e - external, s - stale
Origin codes: i - IGP, e - EGP, ? - incomplete

Network      Nexthop  Metric  LocalPref  Weight  AsPath
-------------  -------  ------  --------  -------  --------
*>i  8.1.1.0/24  0       0       0  65008i
```
From the above outputs, you see that Switch A has not learned a route to AS 65009, and Switch C has learned network 8.1.1.0 but the next hop 3.1.1.2 is unreachable, so the route is invalid.

14. Redistribute connected routes.

15. Configure BGP to redistribute direct routes on Switch B, so that Switch A can obtain the route to 9.1.1.0/24 and Switch C can obtain the route to 3.1.1.0/24.

16. Configure Switch B.

```
switch(config)# router bgp 65009
switch(bgp)# redistribute connected
```

17. Display the BGP routing table on Switch A.

```
switch# show ip bgp

Local AS : 65009
Local Router-id : 1.1.1.1

BGP Table Version : 0
Status codes: * - valid, > - best, i - internal, e - external, s - stale
Origin codes: i - IGP, e - EGP, ? - incomplete

Network  Nexthop  Metric  LocalPref  Weight  AsPath
---------  -------  ------  --------  -------  -----
*<e 2.2.2.2/32  3.1.1.1    0       0       65009?
*<e 3.1.1.0/24  3.1.1.1    0       0       65009?
*<e 8.1.1.0/24  0         0       0       65008i
*<e 8.1.1.0/24  0         0       0       65008i
```

18. Two routes 2.2.2.2/32 and 9.1.1.0/24 have been added in Switch A’s routing table.

19. Display the BGP routing table on Switch C.

```
switch(config)# show ip bgp

Local AS : 65009
Local Router-id : 3.3.3.3

BGP Table Version : 1
Status codes: * - valid, > - best, i - internal, e - external, s - stale
Origin codes: i - IGP, e - EGP, ? - incomplete

Network  Nexthop  Metric  LocalPref  Weight  AsPath
---------  -------  ------  --------  -------  -----
*<e 2.2.2.2/32  9.1.1.1    0      100      I
*<e 3.1.1.0/24  9.1.1.1    0      100      I
*<e 8.1.1.0/24  0         0       0       65008i
*e  9.1.1.0/24  0         0       65008i
*<e 9.1.1.0/24  0         0       65008i
```

20. Route 8.1.1.0 becomes valid with the next hop as Switch A.

21. Verification.

**Route filter configuration**
Network requirements

In the following figure, Switch B establishes eBGP connections with Switch A and C. Configure the No_Export community attribute on Switch A so that routes from AS 10 are not advertised by AS 20 to any other AS.

Figure 19  Network diagram for BGP community configuration

Configuration procedure

1. Configure IP addresses for interfaces (omitted.)

2. Configure eBGP.

3. # Configure Switch A.

```plaintext
switch(config)# router bgp 10
switch(bgp)# bgp router-id 1.1.1.1
switch(bgp)# neighbor 200.1.2.2 remote-as 20
switch(bgp)# network 9.1.1.0/255.255.255.0/8
switch(bgp)# exit
```

4. # Configure Switch B.

```plaintext
switch(config)# bgp 20
switch(bgp)# bgp router-id 2.2.2.2
switch(bgp)# neighbor 200.1.2.1 remote-as 10
switch(bgp)# neighbor 200.1.3.2 remote-as 30
switch(bgp)# exit
```

5. # Configure Switch C.

```plaintext
switch(config)# bgp 30
switch(bgp)# bgp router-id 3.3.3.3
switch(bgp)# neighbor 200.1.3.1 remote-as 20
switch(bgp)# exit
```

6. # Display the BGP routing table on Switch B.

```plaintext
switch(config)# show ip bgp 9.1.1.0
```
7. Switch B advertised routes to Switch C in AS 30.

8. # Display the routing table on Switch C.

```
switch(config)# show ip bgp

Local AS : 30
Local Router-id : 3.3.3.3

BGP Table Version  : 1
Status codes: * - valid, > - best, i - internal, e - external, s - stale
Origin codes: i - IGP, e - EGP, ? - incomplete

Network           Nexthop      Metric  LocalPref  Weight  AsPath
*>i 9.1.1.0/24     200.1.3.1    0       100       10i
```

9. Switch C learned route 9.1.1.0/24 from Switch B.

10. Configure the BGP community.

11. # Configure a routing policy.

```
route-map bgp-out permit seq 10
switch(route-map=bgp-out)# set community no-export
switch(route-map=bgp-out)# exit
```

12. # Apply the routing policy.

```
switch(config)# bgp 10
switch(bgp)# neighbor 200.1.2.2 route-map bgp-out out
```

13. # Display the route on Switch B.

```
switch(config)# show ip bgp 9.1.1.0/24

Local AS : 20 Local Router-id : 2.2.2.2
BGP Table Version : 3

Network : 9.1.1.0/24 Nexthop : 200.1.2.1
Peer : 200.1.2.1 Origin : igp Metric : 0 Local
Pref : Weight : 0 Calc. Local Pref : 100
```
The route 9.1.1.0/24 is not available in the routing table of Switch C.

**BGP route reflector configuration**

**Network requirements**

In the following figure, all switches run BGP.

- There is an eBGP connection between Switch A and Switch B. There are iBGP connections between Switch B and Switch C, and between Switch C and Switch D.
- Switch C is a route reflector with clients Switch B and D.
- Switch D can learn route 1.0.0.0/8 from Switch C.

**Figure 20  Network diagram for BGP route reflector configuration**

**Configuration procedure**

1. Configure IP addresses for interfaces (omitted.)

2. Configure BGP connections.

3. # Configure Switch A.

```
switch(config)# router bgp 100
switch(bgp)# bgp router-id 1.1.1.1
switch(bgp)# neighbor 192.1.1.2 remote-as 200
```

4. # Add network 1.0.0.0/8 to the BGP routing table.

```
switch(bgp)# network 1.0.0.0
switch(bgp)# exit
```

5. # Configure Switch B.
switch(config)# router bgp 200
switch(bgp)# bgp router-id 2.2.2.2
switch(bgp)# neighbor 192.1.1.1 remote-as 100
switch(bgp)# neighbor 193.1.1.1 remote-as 200
switch(bgp)# neighbor 193.1.1.1 next-hop-self
switch(bgp)# exit

6. # Configure Switch C.

switch(config)# router bgp 200
switch(bgp)# bgp router-id 3.3.3.3
switch(bgp)# neighbor 193.1.1.2 remote-as 200
switch(bgp)# neighbor 194.1.1.2 remote-as 200
switch(bgp)# exit

7. # Configure Switch D.

switch(config)# router bgp 200
switch(bgp)# bgp router-id 4.4.4.4
switch(bgp)# neighbor 194.1.1.1 remote-as 200
switch(bgp)# exit

8. Configure the route reflector.

9. # Configure Switch C.

switch(config)# router bgp 200
switch(bgp)# neighbor 193.1.1.2 route-reflector-client
switch(bgp)# neighbor 194.1.1.2 route-reflector-client
switch(bgp)# exit

10. Verify the above configuration.

11. # Display the BGP routing table on Switch B.

switch(config)# show ip bgp

Local AS : 200
Local Router-id : 200.1.2.2

BGP Table Version : 1
Status codes: * - valid, > - best, i - internal, e - external, s - stale
Origin codes: i - IGP, e - EGP, ? - incomplete

<table>
<thead>
<tr>
<th>Network</th>
<th>Nexthop</th>
<th>Metric</th>
<th>LocalPref</th>
<th>Weight</th>
<th>AsPath</th>
</tr>
</thead>
<tbody>
<tr>
<td>*&gt;i 1.0.0.0/24</td>
<td>200.1.3.1</td>
<td>0</td>
<td>0</td>
<td>100i</td>
<td></td>
</tr>
</tbody>
</table>

12. # Display the BGP routing table on Switch D.

switch(config)# show ip bgp

Local AS : 200
Local Router-id : 200.1.2.2
BGP path selection configuration

Network requirements

- In the figure below, all switches run BGP. eBGP connections are between Switch A and Switch B, and between Switch A and Switch C. iBGP connections are between Switch B and Switch D, and between Switch C and Switch D.
- OSPF is the IGP protocol in AS 200.
- Configure the routing policies. Switch D should use the route 1.0.0.0/8 from Switch C as the optimal route.

Figure 21  Network diagram for BGP path selection configuration

<table>
<thead>
<tr>
<th>Device</th>
<th>Interface</th>
<th>IP address</th>
<th>Device</th>
<th>Interface</th>
<th>IP address</th>
</tr>
</thead>
<tbody>
<tr>
<td>Switch A</td>
<td>Vlan101</td>
<td>1.0.0.0/8</td>
<td>Switch D</td>
<td>Vlan400</td>
<td>195.1.1.1/24</td>
</tr>
<tr>
<td></td>
<td>Vlan100</td>
<td>192.1.1.1/24</td>
<td></td>
<td>Vlan300</td>
<td>194.1.1.1/24</td>
</tr>
<tr>
<td></td>
<td>Vlan200</td>
<td>193.1.1.1/24</td>
<td></td>
<td>Vlan400</td>
<td>195.1.1.2/24</td>
</tr>
<tr>
<td>Switch B</td>
<td>Vlan100</td>
<td>192.1.1.2/24</td>
<td></td>
<td>Vlan200</td>
<td>193.1.1.2/24</td>
</tr>
<tr>
<td></td>
<td>Vlan300</td>
<td>194.1.1.2/24</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Configuration procedure
1. Configure IP addresses for interfaces (omitted.)

2. Configure OSPF on Switch B, C, and D.

3. # Configure Switch B.

```
switch(config)# router ospf
switch(ospf)# area 0
switch(ospf)# network 192.1.1.0/ 0.0.0.255
switch(ospf)# network 194.1.1.0/ 0.0.0.255
switch(ospf)# exit
```

4. # Configure Switch C.

```
switch(config)# router ospf
switch(ospf)# enable
switch(ospf)# area 0
switch(ospf)# network 193.1.1.0/ 0.0.0.255
switch(ospf)# network 195.1.1.0/ 0.0.0.255
switch(ospf)# exit
```

5. # Configure Switch D.

```
switch(config)# router ospf
switch(ospf)# enable
switch(ospf)# area 0
switch(ospf)# network 194.1.1.0/ 0.0.0.255
switch(ospf)# network 195.1.1.0/ 0.0.0.255
switch(ospf)# exit
```

6. Configure BGP connections.

7. # Configure Switch A.

```
switch(config)# router bgp 100
switch(bgp)# neighbor 192.1.1.2 remote-as 200
switch(bgp)# neighbor 193.1.1.2 remote-as 200
```

8. # Add network 1.0.0.0/8 to the BGP routing table on Switch A.

```
switch(bgp)# network 1.0.0.0/8
switch(bgp)# exit
```

9. # Configure Switch B.

```
switch(config)# router bgp 200
switch(bgp)# neighbor 192.1.1.1 remote-as 100
switch(bgp)# neighbor 194.1.1.1 remote-as 200
switch(bgp)# exit
```

10. # Configure Switch C.
switch(config)# router bgp 200
switch(bgp)# neighbor 193.1.1.1 remote-as 100
switch(bgp)# neighbor 195.1.1.1 remote-as 200
switch(bgp)# exit

11. # Configure Switch D.

switch(config)# router bgp 200
switch(bgp)# neighbor 194.1.1.2 remote-as 200
switch(bgp)# neighbor 195.1.1.2 remote-as 200
switch(bgp)# exit

12. Configure attributes for route 1.0.0.0/8, making Switch D give priority to the route learned from Switch C.

13. # Configure a higher MED value for the route 1.0.0.0/8 advertised from Switch A to peer 192.1.1.2.

14. # Define a prefix-list to permit route 1.0.0.0/8.

switch(config)# ip prefix-list pl_1 permit 1.0.0.0/24

15. # Define two routing policies, apply_med_50, which sets the MED for route 1.0.0.0/8 to 50, and apply_med_100, which sets the MED for route 1.0.0.0/8 to 100.

switch(config)# route-map apply_med_50 permit
switch(route-map-apply_med_50)# match ip address prefix-list pl_1
switch(route policy)# set metric 50
switch(route policy)# route-map apply_med_50 permit seq 20
switch(route policy)# exit

switch(config)# route-map apply_med_100 permit
switch(route policy)# match ip address prefix-list pl_1
switch(route policy)# set metric 100
switch(route policy)# route-map apply_med_100 permit seq 20
switch(route policy)# exit

16. # Apply routing policy apply_med_50 to the route advertised to peer 193.1.1.2 (Switch C), and apply_med_100 to the route advertised to peer 192.1.1.2 (Switch B.)

switch(config)# bgp 100
switch(bgp)# neighbor 193.1.1.2 route-map apply_med_50 out
switch(bgp)# neighbor 192.1.1.2 route-policy apply_med_100 out
switch(bgp)# exit

17. # Display the BGP routing table on Switch D.

switch(config)# show ip bgp

Local AS : 100
Local Router-id : 194.1.1.1

BGP Table Version : 1

Status codes: * - valid, > - best, i - internal, e - external, s - stale
Origin codes: i - IGP, e - EGP, ? - incomplete
18. You can ensure that route 1.0.0.0/8 is the optimal route. Configure different local preferences on Switch B and C for route 1.0.0.0/8, making Switch D give priority to the route from Switch C.

19. # Define an ip prefix-list on Router C, permitting route 1.0.0.0/8.

```bash
switch(config)# ip prefix-list pl_1 permit 1.0.0.0/8
```

20. # Configure a routing policy named `localpref` on Switch C, setting the local preference of route 1.0.0.0/8 to 200 (the default is 100.)

```bash
switch(config)# route-map localpref permit seq 10
switch(route-policy)# match ip address prefix-list pl_1
switch(route-policy)# set local-preference 200
switch(route-policy)# route-map localpref permit seq 20
```

21. # Apply routing policy `localpref` to routes from peer 193.1.1.1.

```bash
switch(config)# router bgp 200
switch(bgp)# neighbor 193.1.1.1 route-map localpref in
switch(bgp)# exit
```

22. # Display the routing table on Switch D.

```bash
switch(config)# show ip bgp
```

```
Local AS : 100
Local Router-id : 194.1.1.1
BGP Table Version : 1
Status codes: * - valid, > - best, i - internal, e - external, s - stale
Origin codes: i - IGP, e - EGP, ? - incomplete

Network    Nexthop    Metric  LocalPref  Weight  AsPath
-----------------------------------------------
* > e 1.0.0.0/24 194.1.1.2  50      0      100i
* > e 1.0.0.0/24 195.1.1.2 100      0      100i
```

23. You can see that route 1.0.0.0/8 from Switch D to Switch C is the optimal route.

**BGP GR configuration**

**Network requirements**

In the following figure, all switches are BGP switches. There is a eBGP connection between Switch A and Switch B. Switch B and Switch C are connected over an iBGP connection. Enable GR for BGP so that the communication between Switch A and Switch C is not affected when an active/standby main board switchover occurs on Switch B.
Figure 22  Network diagram for BGP GR configuration

![Network diagram for BGP GR configuration](image)

**Configuration procedure**

1. Configure Switch A.

2. # Configure IP addresses for interfaces (omitted.)

3. # Configure the eBGP connection.
   ```
   switch(config)# router bgp 65008
   switch(bgp)# bgp router-id 1.1.1.1
   ```

4. # Configure BGP GR stalepath-timeout (optional.)
   ```
   switch(bgp)# bgp graceful-restart stalepath-time 360
   switch(bgp)# neighbor 200.1.1.1 remote-as 65009
   ```

5. # Add network 8.0.0.0/8 to the BGP routing table.
   ```
   switch(bgp)# network 8.0.0.0/8
   ```

6. # Enable GR for BGP Peer.
   ```
   switch(bgp)# neighbor 200.1.1.1 graceful-restart
   ```

7. Configure Switch B.

8. # Configure IP addresses for interfaces (omitted.)

9. # Configure the eBGP connection.
   ```
   switch(bgp)# router bgp 65009
   ```

10. # Configure BGP GR restart-time and stalepath-timeout (Optional.)
    ```
    switch(bgp)# bgp graceful-restart restart-time 120
    stalepath-time 360
    switch(bgp)# bgp router-id 2.2.2.2
    switch(bgp)# neighbor 200.1.1.2 remote-as 65008
    ```

11. # Configure the iBGP connection.
12. # Configure BGP to redistribute direct routes.
switch(bgp)# redistribute connected

13. # Enable GR capability for BGP Peers.
switch(bgp)# neighbor 9.1.1.2 remote-as 65009
switch(bgp)# neighbor 9.1.1.2 graceful-restart

14. # Configure BGP for non-stop forwarding
switch(bgp)# non-stop

15. Configure Switch C.
16. # Configure IP addresses for interfaces (omitted.)
17. # Configure the iBGP connection.
switch(config)# router bgp 65009
switch(bgp)# bgp router-id 3.3.3.3
switch(bgp)# neighbor 9.1.1.1 remote-as 65009

18. # Configure BGP to redistribute direct routes.
switch(bgp)# redistribute connected

19. BGP Configuration Example
20. # Enable GR for BGP Peer.
switch(bgp)# neighbor 9.1.1.1 graceful-restart

Verification
After completing the above configuration, perform an active/standby switchover on Switch B. Switch A and Switch C should be able to ping each other without any packet drops. Also ensure that there are no flaps of BGP learned routes on the peer switches.

BGP show routines

Synopsis:
show ip bgp [ipv4-addr [mask] [longer-prefixes]]
Displays information about BGP routes installed in the BGP routing information base (RIB.)

ipv4-addr
IP address entered to filter the output to display only a particular host or network in the BGP routing table.

mask
Mask to filter or match hosts that are part of the specified network.

**longer-prefixes**

If a prefix is specified, optionally specify to show routes matching the specified Network/Mask pair only.

```
switch(bgp)# show ip bgp
Local AS : 100    Local Router-id : 10.0.102.138
Status codes: * - valid, > - best, i - internal, e - external, s - stale
Origin codes: i - IGP, e - EGP, ? - incomplete

<table>
<thead>
<tr>
<th>Network</th>
<th>Nexthop</th>
<th>Metric</th>
<th>LocalPref</th>
<th>Weight</th>
<th>AsPath</th>
</tr>
</thead>
<tbody>
<tr>
<td>* e 11.0.0.0/8   10.0.102.40</td>
<td>0</td>
<td>0 200</td>
<td>?</td>
<td></td>
<td></td>
</tr>
<tr>
<td>*&gt;e 11.0.0.0/8   10.0.102.153</td>
<td>0</td>
<td>0 200</td>
<td>i</td>
<td></td>
<td></td>
</tr>
<tr>
<td>*&gt;e 22.0.0.0/8   10.0.102.40</td>
<td>0</td>
<td>0 200</td>
<td>?</td>
<td></td>
<td></td>
</tr>
<tr>
<td>* e 22.0.0.0/8   10.0.102.198</td>
<td>0</td>
<td>0 300</td>
<td>500 ?</td>
<td></td>
<td></td>
</tr>
<tr>
<td>*&gt;e 33.0.0.0/8   10.0.102.40</td>
<td>0</td>
<td>0 200</td>
<td>?</td>
<td></td>
<td></td>
</tr>
<tr>
<td>* e 33.0.0.0/8   10.0.102.198</td>
<td>0</td>
<td>0 300</td>
<td>400 ?</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
```

**Synopsis:**

```
show ip bgp ipv4-address/masklen regexp aspath-reg-ex
```

Displays specific information on the route if the route’s aspath information matches the supplied regular expression. This will filter both on the prefix/len and the regular expression.

```
switch(bgp)# show ip bgp 11.0.0.0/8 regexp 20
Local AS : 100    Local Router-id : 

<table>
<thead>
<tr>
<th>Network</th>
<th>Nexthop</th>
<th>Metric</th>
<th>LocalPref</th>
<th>Weight</th>
<th>AsPath</th>
</tr>
</thead>
<tbody>
<tr>
<td>Network</td>
<td>Nexthop</td>
<td>Metric</td>
<td>LocalPref</td>
<td>Weight</td>
<td>AsPath</td>
</tr>
<tr>
<td>Peer</td>
<td>Origin</td>
<td>valid</td>
<td>yes</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AS-Path</td>
<td>: 200</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

| Communities : 200:20 100:50 |
```

```
switch(bgp)# show ip bgp 11.0.0.0/8 regexp 20
Local AS : 100    Local Router-id : 

<table>
<thead>
<tr>
<th>Network</th>
<th>Nexthop</th>
<th>Metric</th>
<th>LocalPref</th>
<th>Weight</th>
<th>AsPath</th>
</tr>
</thead>
<tbody>
<tr>
<td>Network</td>
<td>Nexthop</td>
<td>Metric</td>
<td>LocalPref</td>
<td>Weight</td>
<td>AsPath</td>
</tr>
<tr>
<td>Peer</td>
<td>Origin</td>
<td>valid</td>
<td>yes</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AS-Path</td>
<td>: 200</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

| Communities : |
```
No AS-Path : 200
Communities : 200:20 100:50

Synopsis:
show ip bgp [ipv4-addr]
Displays all the routes in the IP routing table, including BGP routes.

ipv4-addr
IP address entered to filter the output to display only a particular host or network in the IP routing table.

switch(bgp)# show ip route
IP Route Entries
<table>
<thead>
<tr>
<th>Destination</th>
<th>Gateway</th>
<th>VLAN Type</th>
<th>Sub-Type</th>
<th>Metric</th>
<th>Dist.</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.0.0.0/0</td>
<td>10.0.0.1</td>
<td>static</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>10.0.0.0/16</td>
<td>DEFAULT_VLAN</td>
<td>connected</td>
<td>1</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>11.0.0.0/8</td>
<td>10.0.102.153</td>
<td>bgp</td>
<td>0</td>
<td>20</td>
<td></td>
</tr>
<tr>
<td>22.0.0.0/8</td>
<td>10.0.102.40</td>
<td>bgp</td>
<td>0</td>
<td>20</td>
<td></td>
</tr>
<tr>
<td>33.0.0.0/8</td>
<td>10.0.102.40</td>
<td>bgp</td>
<td>0</td>
<td>20</td>
<td></td>
</tr>
<tr>
<td>99.0.0.0/8</td>
<td>DEFAULT_VLAN</td>
<td>static</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>127.0.0.0/8</td>
<td>reject</td>
<td>static</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>127.0.0.1/32</td>
<td>lo0</td>
<td>connected</td>
<td>1</td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>

Synopsis:
show ip route bgp [ipv4-addr]
Displays only the BGP routes in the IP routing table.

ipv4-addr
IP address entered to filter the output to display only a particular host or network in the BGP routing table.

switch(bgp)# show ip route bgp
IP Route Entries
<table>
<thead>
<tr>
<th>Destination</th>
<th>Gateway</th>
<th>VLAN Type</th>
<th>Sub-Type</th>
<th>Metric</th>
<th>Dist.</th>
</tr>
</thead>
<tbody>
<tr>
<td>11.0.0.0/8</td>
<td>10.0.102.153</td>
<td>bgp</td>
<td>0</td>
<td>20</td>
<td></td>
</tr>
<tr>
<td>22.0.0.0/8</td>
<td>10.0.102.40</td>
<td>bgp</td>
<td>0</td>
<td>20</td>
<td></td>
</tr>
<tr>
<td>33.0.0.0/8</td>
<td>10.0.102.40</td>
<td>bgp</td>
<td>0</td>
<td>20</td>
<td></td>
</tr>
</tbody>
</table>

Synopsis:
show bgp community comm-nums
Displays the list of routes who have specific communities tagged to them.

switch(bgp)# show ip community 200:20 200:30
Local AS : 100
Local Router-id :
Status codes: * - valid, > - best, i - internal, e - external, s - stale
Origin codes: i - IGP, e - EGP, ? - incomplete
Network NextHop Metric LocalPref Weight AsPath
| * e 11.0.0.0/8 10.0.102.40 | 0 | 0 200 ? |
| *e 33.0.0.0/8 10.0.102.40 | 0 | 0 200 ? |

Synopsis:
show ip bgp community regexp community-reg-ex
Displays the routes whose community information matches the supplied regular expression.
show ip bgp community regexp "2"

switch (bgp) # show ip bgp community regexp "2"

Local AS : 100
Local Router-id : 10.0.102.138

Status codes: * - valid, > - best, i - internal, e - external, s - stale
Origin codes: i - IGP, e - EGP, ? - incomplete

<table>
<thead>
<tr>
<th>Network</th>
<th>Nexthop</th>
<th>Metric</th>
<th>LocalPref</th>
<th>Weight</th>
<th>AsPath</th>
</tr>
</thead>
<tbody>
<tr>
<td>* e 11.0.0.0/8 10.0.102.40 0</td>
<td>0</td>
<td>200</td>
<td>?</td>
<td></td>
<td></td>
</tr>
<tr>
<td>* e 11.0.0.0/8 10.0.102.153 0</td>
<td>0</td>
<td>200</td>
<td>i</td>
<td></td>
<td></td>
</tr>
<tr>
<td>* e 22.0.0.0/8 10.0.102.40 0</td>
<td>0</td>
<td>200</td>
<td>?</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Synopsis:**

show ip bgp community comm-num... regexp aspath-reg-ex

Displays the routes whose community information matches the supplied community numbers and also the AS_PATH information matches the supplied regular expression.

switch (bgp) # show ip bgp community 20 regexp "2"

Local AS : 100
Local Router-id : 10.0.102.138

Status codes: * - valid, > - best, i - internal, e - external, s - stale
Origin codes: i - IGP, e - EGP, ? - incomplete

<table>
<thead>
<tr>
<th>Network</th>
<th>Nexthop</th>
<th>Metric</th>
<th>LocalPref</th>
<th>Weight</th>
<th>AsPath</th>
</tr>
</thead>
<tbody>
<tr>
<td>* e 11.0.0.0/8 10.0.102.40 0</td>
<td>0</td>
<td>200</td>
<td>?</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Synopsis:**

show ip bgp community comm-num... exact regexp aspath-reg-ex

Displays the routes whose community information matches exactly the supplied community numbers and also whose AS_PATH information matches the supplied regular expression.

switch (bgp) # show ip bgp community 200:20 100:50 exact regexp "2"

Local AS : 100
Local Router-id : 10.0.102.138

Status codes: * - valid, > - best, i - internal, e - external, s - stale
Origin codes: i - IGP, e - EGP, ? - incomplete

<table>
<thead>
<tr>
<th>Network</th>
<th>Nexthop</th>
<th>Metric</th>
<th>LocalPref</th>
<th>Weight</th>
<th>AsPath</th>
</tr>
</thead>
<tbody>
<tr>
<td>* e 11.0.0.0/8 10.0.102.40 0</td>
<td>0</td>
<td>200</td>
<td>?</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Synopsis:**

show ip bgp regexp reg-ex

Displays all routes whose AS_PATH matches the regular-expression given.

switch (bgp) # show ip bgp regexp "^300"

Local AS : 100
Local Router-id : 10.0.102.138

Status codes: * - valid, > - best, i - internal, e - external, s - stale
Origin codes: i - IGP, e - EGP, ? - incomplete

<table>
<thead>
<tr>
<th>Network</th>
<th>Nexthop</th>
<th>Metric</th>
<th>LocalPref</th>
<th>Weight</th>
<th>AsPath</th>
</tr>
</thead>
<tbody>
<tr>
<td>* e 22.0.0.0/8 10.0.102.198 0</td>
<td>0</td>
<td>300</td>
<td>500</td>
<td>?</td>
<td></td>
</tr>
<tr>
<td>* e 33.0.0.0/8 10.0.102.198 0</td>
<td>0</td>
<td>300</td>
<td>400</td>
<td>?</td>
<td></td>
</tr>
</tbody>
</table>

**Synopsis:**

show ip bgp [ipv4-addr/masklen [longer-prefix]] route community
Displays basic route information (destination and nexthop) and the communities tagged to the route in full. This show routine is especially helpful when you want to look at the communities that are tagged to all routes at a glance.

```
switch(bgp)# show ip bgp 22.0.0.0/8 route community

Local AS : 100                Local Router-id : 

Status codes: * - valid, > - best, i - internal, e - external, s - stale
Origin codes: i - IGP, e - EGP, ? - incomplete

Network    Nexthop    Community
<-----------------------------
*<e 22.0.0.0/8 10.0.102.40  200:20 100:50    
*e 22.0.0.0/8 10.0.102.198 no-export
```

**Synopsis:**

```
show ip bgp neighbor [ipv4-addr]
```

Displays information about the state of BGP’s IPv4 peering sessions.

```
switch(bgp)# show ip bgp neighbor 10.0.102.40

BGP Neighbor 10.0.102.40

BGP Version : 4
Remote Router ID : 10.0.102.40 Local Router ID :10.0.102.138
Remote-AS : 200       Local-AS : 100
Remote Port : 179       Local Port : 56126
State : Established       Up Time : 0h:3m:29s
Admin Status : Start       Link Type : External
Conn Established : 1       Conn Dropped : 0
Last Read : 0h:0m:29s       Last Write : 0h:0m:29s
Last reset time : 0h:0m:0s       Error Subcode Sent : 0
Last reset reason: Never       Gr. Restart Time : 120 secs.

MAXIMUM Prefix : 4294967295   Send Community : Yes
Weight : 0             RtReflectorClient : No
Use MED : Yes            Passive : No
AS-Override : No        Allow-AS in : 0
Ignore Lead AS : No       Out-Delay : 0
Remove Private AS : No     Ttl : 1
Update Source :
Route-Map-In :
Route-Map-Out :
Password :

Capability                  Announced    Received
-----------------------------    ---------    ---------
Route Refresh                No   Yes
Dynamic                     No    No
Graceful Restart (ipv4-uni) Yes    No
Multi-protocol (ipv4-uni) Yes     Yes

Message Type     Sent    Received
-----------------------------    ---------    ---------
Opens                  1       1
Notifications         0       0
Capability              0       0
Updates                 1       1
Keepalives             4       4
Route Refresh          0       0
```
Total 6 6
Prefix Activity Sent Received
----------------------------------- ------ ------
Prefixes Current 1 3
Prefixes Total 1 3
Implicit Withdraw 0 0
Explicit Withdraw 0 0
Used as BestPath n/a 2

Local Policy Denied Prefixes Outbound Inbound
----------------------------------- ------ ------
Routemap 0 0
Bad lead AS n/a 0
Exceeded Max-prefix n/a 0
Exceeded Allow-as in n/a 0
Total 0 0

Max Min
Number of NLRIs in the update sent 1 0

Synopsis:
show ip bgp as-path
Displays the list of AS_PATHs that BGP has learned from the routing information it has received.

switch# show ip bgp as-path
BGP AS-Path Information

<table>
<thead>
<tr>
<th>AS Path</th>
<th>Metric</th>
<th>RefCount</th>
</tr>
</thead>
<tbody>
<tr>
<td>I</td>
<td>0</td>
<td>4</td>
</tr>
<tr>
<td>?</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td>200 i</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>300 ?</td>
<td>250</td>
<td>2</td>
</tr>
</tbody>
</table>

Synopsis:
show ip bgp redistribute
Displays the list of protocols whose routes are being redistributed into BGP.

switch# show ip bgp redistribute
Route type RouteMap
--------- ------------------------
static rtmap-static
rip

Synopsis:
show ip bgp summary
Displays a summarized view of global BGP configuration and current BGP neighbor peering state.

**BGP solution use cases**

**Solution 1 – Campus iBGP**

Two use cases are presented. The first illustrates the extension of BGP into an Enterprise routing environment. The second case shows BGP connectivity in a remote site environment.
Figure 23 Solution 1 – Campus iBGP

Devices
A
WAN Gateway Router
B
Enterprise Core Router
C
Enterprise Core Router (Campus Edge)
D
Campus Core Routing Switch
E
Campus Distribution Routing Switch
F
Edge Switch

In the figure above, multiple campus domains are segmented by using BGP in the Enterprise core. Traditionally, solutions have been used with devices E and F, facing the client or server network edges. With the introduction of BGP functionality, it becomes possible to position solutions at locations B, C, and D.

With proper filtering, a routing switch with 20,000 routes can be used in an iBGP deployment. A device at location C represents the boundary between interior gateway protocol (IGP) domains, and the BGP core. Functionality used on this device includes redistribution with route maps and the establishment of BGP communities. Devices at location B require AS path filtering. All locations within the BGP AS require the remaining “Foundation” features (Route Reflection, Refresh, Multihop, and so forth)

Additional Autonomous Systems may be configured within a network, resembling the Enterprise core module as shown in the diagram. With larger Enterprise customers, it is likely that an AS that is directly adjacent to IGP campus modules will be the location for foundation BGP solutions. See Figure 24.
**Figure 24** Multiple internal AS deployments with Campus iBGP solution

The core routing switch (device C) can establish eBGP peering with the Enterprise Core. It is possible to utilize the foundation Campus iBGP feature to satisfy some of these solutions.

A  
Enterprise Core Router

B  
Enterprise Core Router (Campus Edge)

C  
Campus Core Routing Switch

D  
Campus Distribution Routing Switch (or Collapsed Core)

E  
Edge Switch

**Solution 2 – Remote site iBGP**
You have the alternative of using static routes or BGP to connect to your service provider. For multi-homing or policy control, you can choose to deploy BGP. This may be used for internet connectivity. Foundation iBGP solutions do not carry full internet routing tables, so the diagram above requires that 1) only default routes are taken from the internet and 2) multiple VRF instances do not exist at a single physical remote site.

The deployment of device A may require additional traffic shaping and scalability features. If you prefer extending BGP routing to devices B or C, you can use BGP functionality on a routing switch. In this deployment model, the routing switch would be used for route redistribution and the marking of communities.

**Troubleshooting BGP**

**Debug log messages**

1. Logs per-peer BGP State Transitions.
2. Logs per-peer arrivals of a new BGP update.
3. Logs per-peer Time-outs (Hold-time, Graceful Restart Timeout.)
4. Logs Memory problems in case buffer-allocations fail.

**No BGP peer relationship established**
Symptom
Display BGP peer information using the `show ip bgp neighbor` command. A connection to a peer has not been established.

Analysis
To become BGP peers, any two routers need to establish a TCP session using port 179 and exchange open messages successfully.

Solution
1. Use the `show ip bgp neighbor` command to verify the peer's IP address.
2. If the loopback interface is used, check whether the `neighbor connect interface` command is configured.
3. If the peer is a non-direct eBGP peer, check whether the `neighbor ebgp multihop` command is configured.
4. Check whether a route to the peer is available in the routing table.
5. Use the `ping` command to check connectivity.
6. Use the
   
   ```
   display
tcp status
   ```
   command to check the TCP connection.
7. Check whether an ACL is configured that disables TCP port 179.
Bidirectional Forwarding Detection (BFD) is intended for use only on v3 modules.

Bidirectional Forwarding Detection (BFD) is a low-overhead, short-duration method for detection of failures in the path between adjacent forwarding engines, including the interfaces, data link(s), and, to the extent possible, the forwarding engines themselves. It also provides a single mechanism that can be used for liveness detection between a pair of devices over any media, at any protocol layer, with a wide range of Detection Times and overhead, to avoid proliferation of different methods.

**Asynchronous mode:** In Asynchronous mode, an operating device periodically sends BFD control packets. If the device does not receive BFD control packet from the peer within the specified interval, it tears down the BFD session.

**Echo mode:** In Echo mode, an operating device periodically sends BFD echo packets. The peer device returns the received BFD echo packets back without processing them. If the sending device does not receive BFD echo packet from the peer within the specified interval, the session is considered down.

### Commands

#### Per-session command VLAN

All configuration commands described in this section belong to VLAN context. That is, the configuration will be applied to all the sessions under the VLAN identified by the VLAN ID.

**Set intervals**

This command helps to assign the minimum transmit interval and minimum receive interval in the range 1 to 20 seconds. Detect multiplier value is assigned as a number between 1 and 5. By default, the minimum transmit and receive interval is 3 seconds and multiplier value is 5.

**Syntax**

```
bfd min-transmit-interval TXSECONDS min-receive-interval RXSECONDS detect-multiplier MULTIPLIER
```

**Description**

Configure Bidirectional Forwarding Detection (BFD) for the VLAN.

**Options**

- **min-transmit-interval**
  Update the minimum transmit interval of the BFD session.

- **min-echo-receive-interval**
  Update the minimum echo receive interval of the BFD session.

- **authentication**
  Configure authentication mode and key for all BFD sessions under the current VLAN.

**Syntax**

```
bfd authentication | min-echo-receive-interval | min-transmit-interval
```

**Description**

Configure Bidirectional Forwarding Detection (BFD) for the VLAN.
Update BFD timer intervals for all the sessions under the current VLAN.

**Options**

**min-transmit-interval**
Update the minimum transmit interval of the BFD session.

**min-receive-interval**
Update the minimum receive interval of the BFD session.

**detect-multiplier**
Update the detect multiplier count of the BFD session.

**txseconds**
The time interval, in the range 1 to 20 seconds, between the transmission of two BFD hello packets.

**rxseconds**
The time interval, in the range 0 to 20 seconds, between the reception of two BFD hello packets.

0 indicates the local end is not interested in receiving hello packets from the peer.

**multiplier**
Number of BFD packets, in the range 1 to 5, that are allowed to be missed before BFD session times out.

- If min-transmit-interval or min-receive-interval value is configured as 1 sec, the value of detect multiplier should be at least 3.
- If detect multiplier value is 1, the value of min-transmit-interval and min-receive-interval should be at least 3 sec.

---

**Set intervals configuration**

```bash
; J9850A Configuration Editor; Created on release #KB.16.02.0000x
; Ver #0b:fc.59.f4.7b.ff.fc.ff.ff.3f.ef:0d
hostname "switch"
module A type j9989a
module C type j9550a
module F type j9987a
snmp-server community "public" unrestricted
noobm
  ip address dhcp-bootp
exit
vlan 1
  name "DEFAULT_VLAN"
  untagged A2-A24,C1-C24,F1-F24
  ip address dhcp-bootp
exit
ip routing
router ospf
  area 0.0.0.2
  area 0.0.0.3
  area backbone
  enable
exit
bfd enable
bfd echo-src-ip-address 2.2.2.2
vlan 20
  untagged A1
  bfd min-transmit-interval 10 min-receive-interval 10 detect-multiplier 3
  ip address 100.100.100.100 255.255.255.0
  ip ospf 100.100.100.100 area backbone
```
ip ospf 100.100.100.100 bfd
exit

Show BFD session

Switch# show bfd-session 1

BFD Session Information - Session 1

- Min Tx Interval (sec) : 10
- Min Rx Interval (sec) : 10
- Min Echo Rx Interval (msec) : 500
- Detect Multiplier : 3
- Application : OSPF
- Local Discriminator : 1
- Remote Discriminator : 1
- Echo : Enabled
- Local Diagnostic : No diagnostics configured.

VLAN Source IP Destination IP State Pkt In Pkt Drop Pkt Out
---- 100.100.100.100 100.100.100.101 Up 322 0 320

Echo intervals

This command helps to assign the minimum receive interval of echo session, either 0 or in the range 50 to 1000 milliseconds. The default interval is 500 milliseconds. Zero indicates that the local end is not interested in receiving echo packets from the peer.

Syntax

```
bfd min-echo-receive-interval MILLISECONDS
```

Description

Update the minimum receive interval for echo packets of all the sessions under the current VLAN. When minimum echo receive interval is set to 0 milliseconds for the BFD session under OSPF, incoming BFD echo packets are not processed. When minimum echo receive interval is set to 0 milliseconds for the BFD session under VRRP, the default interval 500 milliseconds is considered.

Options

- **min-echo-receive-interval**
  Update minimum echo interval of the BFD session.

- **milliseonds**
  The time interval, either 0 or in the range 50 to 1000 milliseconds, between the reception of two BFD echo packets. 0 indicates that the local end is not interested in receiving echo packets from the peer.

Echo intervals configuration

```bash
; J9850A Configuration Editor; Created on release #KB.16.02.0000x
; Ver #0b:fc.59.f4.7bf.ff.f.3f.ef:0d
hostname "switch-name"
module A type j9889a
module C type j9550a
module F type j9987a
snmp-server community "public" unrestricted
oobm
  ip address dhcp-bootp
exit
```
Show BFD-session

Switch# show bfd-session 1

BFD Session Information - Session 1

Min Tx Interval (sec) : 3
Min Rx Interval (sec) : 3
Min Echo Rx Interval (msec) : 700
Detect Multiplier : 5
Authentication Mode : NONE
Password : ""
Application : OSPF
Local Discriminator : 1
Remote Discriminator : 1
Echo : Enabled
Local Diagnostic : No diagnostics configured.

<table>
<thead>
<tr>
<th>VLAN</th>
<th>Source IP</th>
<th>Destination IP</th>
<th>State</th>
<th>Pkt In</th>
<th>Pkt Drop</th>
<th>Pkt Out</th>
</tr>
</thead>
<tbody>
<tr>
<td>20</td>
<td>100.100.100.100</td>
<td>100.100.100.101</td>
<td>Up</td>
<td>322</td>
<td>0</td>
<td>320</td>
</tr>
</tbody>
</table>

Enable BFD under OSPF

This command helps to enable BFD under Open Shortest Path First (OSPF) for a particular IP (VLAN specific). When OSPF adjacency with a neighbor attains state FULL, BFD is informed to create a session in asynchronous mode. After the BFD session is UP, echo is enabled for the session.

Syntax

ip ospf IP-ADDR bfd
no ip ospf IP-ADDR bfd

Description

Enable BFD in OSPF for VLAN specific IP address.

Options

IP-ADDR

Specify the IP address of VLAN for which BFD has to be enabled.

BFD

Configure Bidirectional Forwarding Detection (BFD) for the VLAN.
Both end-points hosting the BFD sessions must be on the same network segment and in the same area.

Enable BFD under OSPF configuration

```
; J9850A Configuration Editor; Created on release #KB.16.02.0000x
; Ver #0b:fc.59.f4.7b.ff.fc.ff.fc.fc.fc.fc.fc.fc:0d
hostname "switch"
module A type j9989a
module C type j9550a
module F type j9987a
snmp-server community "public" unrestricted
  oobm
    ip address dhcp-bootp
  exit
vlan 1
  name "DEFAULT_VLAN"
  untagged A2-A24,C1-C24,F1-F24
ip address dhcp-bootp
exit
ip routing
router ospf
  area 0.0.0.2
  area 0.0.0.3
  area backbone
  enable
  exit
bfd enable
bfd echo-src-ip-address 2.2.2.2
vlan 20
  untagged A1
  bfd min-transmit-interval 10 min-receive-interval 10 detect-multiplier 3
  bfd min-echo-receive-interval 700
  ip address 100.100.100.100 255.255.255.0
  ip ospf 100.100.100.100 area backbone
  ip ospf 100.100.100.100 bfd
  exit
```

Show BFD-session

```
switch# show bfd-session 1

BFD Session Information - Session 1

  Min Tx Interval (sec) : 10
  Min Rx Interval (sec) : 10
  Min Echo Rx Interval (msec) : 700
  Detect Multiplier : 3
  Authentication Mode : NONE
  Password : ""
  Application : OSPF
  Local Discriminator : 1
  Remote Discriminator : 1
  Echo : Enabled
  Local Diagnostic : No diagnostics configured.

  VLAN Source IP  Destination IP  State Pkt In Pkt Drop Pkt Out
  ----  ---------  -----------  -------  -------  -------  -------
  20 100.100.100.100 100.100.100.101 Up 322 0 320
```

Enable BFD under VRRP
This command allows the user to enable BFD under Virtual Router Redundancy Protocol (VRRP). BFD asynchronous mode is not supported for VRRP. Only an echo session will be initiated from VRRP backup to the VRRP main router for a given VR instance in a given VLAN.

Syntax

```plaintext
vrrp vrid VR-ID bfd IP-ADDR
no vrrp vrid VR-ID bfd IP-ADDR
```

Description

Enable BFD in VRRP for VLAN specific IP address.

Options

**BFD**

Configure Bidirectional Forwarding Detection (BFD) for the VLAN.

**IP-ADDR**

Configure the IP address of the peer to enable BFD for the VR.

---

**NOTE**

BFD for VRRP is applicable only for two-router redundant systems. Only one BFD-VRRP session will be maintained for the multiple VRIDs configured on any specific VLAN. BFD-VRRP session is unique for multiple VRIDs enabled with BFD.

---

**Enable BFD under VRRP Configuration**

```plaintext
; J9850A Configuration Editor; Created on release #KB.16.02.0000x
; Ver #0b:fc.59.f4.7b.ff.fc.ff.3f.ef:0d
hostname "switch"
module A type j9989a
module C type j9550a
module F type j9987a
snmp-server community "public" unrestricted
  oobm
  ip address dhcp-bootp
  exit
vlan 1
  name "DEFAULT_VLAN"
  untagged A2-A24,C1-C24,F1-F24
  ip address dhcp-bootp
  exit
ip routing
router vrrp
  ipv4 enable
  exit
bfd enable
bfd echo-src-ip-address 2.2.2.2
vlan 10
  untagged A
  bfd min-transmit-interval 10 min-receive-interval 10 detect-multiplier 3
  bfd min-echo-receive-interval 700
  ip address 100.100.100.100 255.255.255.0
vrrp vrid 7
  virtual-ip-address 100.100.100.102
  priority 255
  bfd 100.100.100.102
  enable
  exit
  exit
```

**Show BFD**
switch# sh bfd

Bidirectional Forwarding Detection (BFD) Information

Admin Status : Enabled
Echo source IP : 2.2.2.2

Global Statistics:
Total Number of Control Packets Transmitted : 5
Total Number of Control Packets Received : 5
Total Number of Control Packets Dropped : 0

<table>
<thead>
<tr>
<th>Session VLAN Source IP</th>
<th>Dest IP</th>
<th>Echo</th>
<th>State</th>
<th>Application</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>10</td>
<td>100.100.100.100</td>
<td>100.100.100.102</td>
<td>Enabled Up</td>
</tr>
</tbody>
</table>

switch# show bfd 1

BFD Session Information

Min Echo Rx (in msecs) : 700

<table>
<thead>
<tr>
<th>Session VLAN Source IP</th>
<th>Destination IP</th>
<th>Echo</th>
<th>State</th>
<th>Application</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>10</td>
<td>100.100.100.100</td>
<td>100.100.100.102</td>
<td>Enabled Up</td>
</tr>
</tbody>
</table>

Set BFD authentication mode and password

This command allows to specify authentication mode and key to be shared with BFD peer for all sessions under VLAN context.

Syntax

 bfd authentication keyed-sha1 | meticulous-Keyed-sha1 KEY-ID key simple | encrypted password
no bfd authentication keyed-sha1 | meticulous-Keyed-sha1 KEY-ID key simple | encrypted password

Description

Configure authentication mode and key for all BFD sessions under the current VLAN.

Options

BFD

Configure Bidirectional Forwarding Detection (BFD) for the VLAN.

authentication

Configure authentication mode and key for all BFD sessions under the current VLAN.

Keyed-sha1

Use authentication mode SHA-1.

Meticulous keyed-sha1

Use authentication mode meticulous SHA-1.

Key-id

Specify the ID, in the range 0 to 255, to uniquely recognize a key.

key

Enter the password to be shared between BFD peers.

simple

Configure the authentication password using a plaintext string.
encrypted

Configure the authentication password using a pre-encrypted string copied from a compatible HP networking device.

Password will be prompted interactively as above and set the entered value in the configuration.

**Simple password**

```
switch(vlan-10)# bfd authentication keyed-shal 1 key simple
Enter password#: ******
Re-enter password#: ******
```

**Without include or encrypt credentials:**

```
; J9850A Configuration Editor; Created on release #KB.16.02.0000x
; Ver #0b:fc.59.f4.7b.ff.ff.fc.ff.3f.ef:0d
hostname "switch-name"
module A type j9989a
module C type j9950a
module F type j9987a
snmp-server community "public" unrestricted
 oobm
  ip address dhcp-bootp
  exit
vlan 1
  name "DEFAULT_VLAN"
  untagged A2-A24,C1-C24,F1-F24
  ip address dhcp-bootp
  exit
ip routing
router ospf
  area 0.0.0.2
  area 0.0.0.3
  area backbone
  enable
  exit
bfd enable
bfd echo-src-ip-address 2.2.2.2
vlan 20
  untagged A1
    bfd min-transmit-interval 10 min-receive-interval 10 detect-multiplier 3
    bfd min-echo-receive-interval 700
    bfd authentication meticulous-Keyed-shal 1 key simple
  ip address 100.100.100.100 255.255.255.0
  ip ospf 100.100.100.100 area backbone
  ip ospf 100.100.100.100 bfd
  exit
```

**With include credentials:**

```
; J9850A Configuration Editor; Created on release #KB.16.02.0000x
; Ver #0b:fc.59.f4.7b.ff.ff.fc.ff.3f.ef:0d
hostname "switch-name"
module A type j9989a
module C type j9950a
module F type j9987a
include-credentials
snmp-server community "public" unrestricted
 oobm
  ip address dhcp-bootp
  exit
vlan 1
```
name "DEFAULT_VLAN"
untagged A2-A24,C1-C24,F1-F24
ip address dhcp-bootp
exit
ip routing
router ospf
  area 0.0.0.2
  area 0.0.0.3
  area backbone
  enable
  exit
  bfd enable
  bfd echo-src-ip-address 2.2.2.2
vlan 20
  untagged A1
  bfd min-transmit-interval 10 min-receive-interval 10 detect-multiplier 3
  bfd min-echo-receive-interval 700
  bfd authentication meticulous-Keyed-shal 1 key simple "hp1234"
  ip address 100.100.100.100 255.255.255.0
  ip ospf 100.100.100.100 area backbone
  ip ospf 100.100.100.100 bfd
  exit
With include and encrypt credentials:

; J9850A Configuration Editor; Created on release #KB.16.02.0000x
; Ver #0b:fc.59.f4.7b.ff.ff.fc.ff.3f.ef:0d
hostname "switch-name"
module A type j9989a
module C type j9550a
module F type j9987a
encrypt-credentials
include-credentials
snmp-server community "public" unrestricted
oobm
  ip address dhcp-bootp
  exit
vlan 1
  name "DEFAULT_VLAN"
  untagged A2-A24,C1-C24,F1-F24
  ip address dhcp-bootp
  exit
ip routing
router ospf
  area 0.0.0.2
  area 0.0.0.3
  area backbone
  enable
  exit
  bfd enable
  bfd echo-src-ip-address 2.2.2.2
vlan 20
  untagged A1
  bfd min-transmit-interval 10 min-receive-interval 10 detect-multiplier 3
  bfd min-echo-receive-interval 700
  bfd authentication meticulous-Keyed-shal 1 key simple
  aH4ihIbkKOGNXpHneZEJqVRuqigYDxOhLCh0TDTpJUA="
  ip address 100.100.100.100 255.255.255.0
  ip ospf 100.100.100.100 area backbone
  ip ospf 100.100.100.100 bfd
  exit

How to input encrypted password
Design considerations for BFD authentication

Supported BFD authentication modes

As per section 6.7 of RFC 5880, implementations supporting authentication MUST support both types of SHA1 authentication. Other forms of authentication are optional. For the first release, **only** Keyed SHA1 and Meticulous Keyed SHA1 authentication schemes will be supported as per the RFC requirement.

Multiple authentication keys on a specific VLAN can not be configured. Each VLAN can have only one Authentication key to be configured.

Configuration Requirements

The following table lists the actions during show run and download mode considering include/encrypt credentials.

| Maximum of 64 BFD sessions are supported. |
**show run output**

<table>
<thead>
<tr>
<th>BFD authentication password will not be displayed.</th>
<th>Reboot with saved config</th>
</tr>
</thead>
<tbody>
<tr>
<td>■ Include credentials: Disabled</td>
<td>The plain-text password in the config will be used to</td>
</tr>
<tr>
<td>■ Encrypt credentials: Disabled</td>
<td>update the protocol data structures.</td>
</tr>
<tr>
<td>■ Include credentials: Disabled</td>
<td>■ Download config file: Password ignored</td>
</tr>
<tr>
<td>■ Encrypt credentials: Disabled</td>
<td></td>
</tr>
</tbody>
</table>

| BFD authentication password will not be displayed.                      | The encrypted password in the config will be decrypted        |
|------------------------------------------------------------------------| used to update the protocol data structures.                  |
| ■ Include credentials: Disabled                                        | ■ Download config file: Password ignored                      |
| ■ Encrypt credentials: Enabled                                          |                                                               |

| BFD authentication password will be displayed in plaintext.              | The plaintext password in the config will be used to         |
| ■ Include credentials: Enabled                                          | update the protocol data structures.                         |
| ■ Encrypt credentials: Disabled                                         | ■ BFD authentication password stored as plaintext.           |
| ■ Include credentials: Enabled                                          |                                                               |

| BFD authentication password will be displayed in encrypted form.        | The encrypted password in the config will be decrypted        |
| ■ Include credentials: Enabled                                          | used to update the protocol data structures.                  |
| ■ Encrypt credentials: Enabled                                          | ■ BFD authentication password stored as encrypted.            |
| ■ Include credentials: Enabled                                          |                                                               |

**BFD static routing**

BFD static routing is available only on switches running KB software. BFD over IPv6 static routes is not supported.

Bidirectional Forwarding Detection (BFD) provides short-duration detection of failures in the path between adjacent forwarding devices. You can now associate BFD with static routes to monitor the reachability of the next-hop gateway. When BFD is configured over a static route, it monitors the connectivity of the local router with the next hop IP. BFD must be configured on the local router interface as well as the next-hop router interface.

If a failure occurs, the corresponding BFD session is taken down and the corresponding static route entry from the Routing Information Base (RIB) is removed. If an alternate route to the destination exists, it is automatically added to the RIB. When the BFD session goes down, the session is not deleted, but the reachability of the next hop is attempted periodically by sending periodic BFD control packets. Once the next hop is reachable, BFD changes the session state to UP and installs the corresponding route in the RIB.

If the static route already exits, BFD can be configured on top of it. Nothing changes. If the static route does not exist, it can be configured along with the BFD configuration.

The **ip route bfd** command helps to enable BFD under ip static route for a particular next-hop destination and BFD source IP.

When BFD is configured over a static route, BFD starts monitoring the connectivity of the local router with the next hop ip. Static routes are added to or removed from the Routing Information Base (RIB) based on the status of BFD sessions. To successfully establish a BFD session, BFD should be configured on the local router interface as well as the next-hop router interface. BFD starts monitoring a given static route once the corresponding BFD session reaches the UP state.

Connectivity to the next hop router may be lost due to an event like an interface down or a neighbor going down; in such a scenario, BFD can detect such failures and trigger corrective measures to reduce network outages. When BFD detects a link failure (that is, BFD does not receive a control or echo packet for a specified amount of time), it
takes the corresponding BFD session down and removes the corresponding static route entry from the RIB. If an alternate route to destination exists, it is automatically added to the RIB.

When the BFD session goes down, the session is not deleted. Once the next hop is reachable, BFD changes the session state to UP and installs the corresponding route in the RIB.

The BFD session is maintained in the BFD session database until it is explicitly removed by the user. Static route BFD is not supported for monitoring multi-hop connectivity.

**ip route bfd**

**Syntax**

```
ip route <destination network A.B.C.D> {<subnet mask A.B.C.D> | /<prefix length>} {<next hop A.B.C.D> | vlan <VLAN ID>} {bfd source-ip <IP address A.B.C.D> | bfd destination-ip <IP address A.B.C.D>} 
no ip route <destination network A.B.C.D> {<subnet mask A.B.C.D> | /<prefix length>} {<next hop A.B.C.D> | vlan <VLAN ID>} {bfd source-ip <IP address A.B.C.D> | bfd destination-ip <IP address A.B.C.D>}
```

**Description**

Enables bidirectional forwarding detection under IP static routing.

The no form disables bidirectional forwarding detection in the specified static route. The no form removes only the BFD configuration provided the bfd option is given. Otherwise, the entire static route along with the BFD configuration is removed.

<table>
<thead>
<tr>
<th>NOTE</th>
</tr>
</thead>
<tbody>
<tr>
<td>A maximum of 64 BFD sessions (shared between OSPF, VRRP, and STATIC) are supported. A maximum of 16 static route BFD sessions is supported. All other BFD parameters (like detect multiplier, transmit/receive intervals, authentication) are automatically obtained from the associated VLAN interface of the bfd source-ip.</td>
</tr>
</tbody>
</table>

**Parameters**

- **destination network A.B.C.D**
  IP address mask of the destination network.

- **subnet mask A.B.C.D**
  IP address of the subnet mask.

- **prefix length**
  Network mask length for the destination.

<table>
<thead>
<tr>
<th>NOTE</th>
</tr>
</thead>
<tbody>
<tr>
<td>The length parameter must be preceded by / (forward slash).</td>
</tr>
</tbody>
</table>

- **next hop A.B.C.D**
  IP address of the next hop.

- **vlan**

- **vlan-id**
  Specifies the destination VLAN for this route.

- **bfd source-ip IP address A.B.C.D**
  Specifies the local router source IP, which sends BFD packets to the next hop destination in order to monitor connectivity with it. This option is required when the next hop IP is specified in the IP route command.

- **bfd destination-ip IP address A.B.C.D**
  Specifies the next hop destination IP to which BFD packets are sent from the local source IP configured on a specified VLAN. This option is required when the destination VLAN is specified in the IP route command.

**Restrictions**
A maximum of 16 unique (src/dst combinations) static route BFD sessions is supported.

A single Static Route BFD session can be shared with multiple static routes having the same next-hop IP and BFD source IP. It is recommended that you keep the total number of static routes, whose next-hop shares the same physical link, under 64.

BFD is supported on single-hop ipv4 static routes. Multi-hop functionality is not supported. BFD neighbors must be no more than one IP hop away for Echo mode.

BFD for Static Route is not supported over management VLAN. BFD source IP address should not be DHCP learned.

BFD authentication mismatches can cause the registered applications (OSPF and STATIC) to flap.

BFD is resource intensive protocol. Setting aggressive control timers for static routes further impacts the system, which could lead to session flaps. It is recommended that a transmit interval be a minimum of 2. Instead, use Echo mode to achieve faster failure detection.

BFD is only a failure detection protocol. As the number of routes increases, there can be a slight increase in the route convergence times, even though the failure detection times do not change.

If BFD Echo enters the disabled state on any session, it remains there. After fixing the connectivity issues, the administrative state must be toggled to re-enable it.

Once a static route BFD session is UP, any change in BFD authentication causing mismatch for the static route BFD session results in a complete removal of the static route from the RIB until the BFD authentication mismatch is resolved. If a mismatch exists before the session is established, the session remains DOWN. The route is undisturbed. BFD can take action only if the session is UP and then transitions to DOWN or vice-versa.

Usage
There are two distinct ways to configure the static route. BFD configuration is allowed on each type of static route configuration:

- Method 1: Configure the gateway as the IP-address of the next-hop device.
- Method 2: Configure the gateway as the `vlan-id` to which the next-hop device is connected.

Example
Configuring static route BFD using next hop IP and BFD source IP. All other BFD parameters (like detect multiplier, transmit/receive intervals, authentication) are automatically obtained from the associated VLAN interface of the `bfd source-ip`. In the following example, the BFD parameters are obtained from VLAN 10.

Switch 1:

```
ip route <destination-network/prefix-length> <gateway-ip> bfd {source-ip | destination-ip} <IP-Address-on-self-device>
```

BFD is enabled between self-device and the peer-device for the static route configured with gateway as Nexthop-IP-Address (Method 1):

```
switch(config)# ip route 172.192.4.0/24 172.16.4.2 bfd source-ip 172.16.4.1
switch(config)#
switch(config)# show running-config
```

```
Running configuration:

; J9850A Configuration Editor; Created on release #KB.16.04.0000x
; Ver #10:1b.7f.bf.bb.fb.7c.59.fc.7b.ff.ff.fc.3f.3f.3f.01
hostname "switch"
module A type j9992a
ip route 172.192.4.0 255.255.255.255.0 172.16.4.2 bfd source-ip 172.16.4.1
ip routing
```
snmp-server community "public" unrestricted
  oobm
  ip address dhcp-bootp
  exit
bfd enable
bfd echo-src-ip-address 2.2.2.2
vlan 1
  name "DEFAULT_VLAN"
  no untagged A1
  untagged A2-A21
  ip address dhcp-bootp
  exit
vlan 10
  name "VLAN10"
  untagged A1
  bfd min-echo-receive-interval 700
  ip address 172.16.4.1 255.255.255.0
  exit
no allow-v2-modules

switch(config)# show bfd

Bidirectional Forwarding Detection (BFD) Information

  Administrative Status : Enabled
  Echo Source IP : 2.2.2.2

  Maximum number of sessions supported : 128
  Number of sessions reserved for internal use : 0
  Total Number of Sessions : 1
  Number of Sessions Up : 1
  Number of Sessions Down : 0

  Global Statistics:
  Total Number of Control Packets Transmitted : 9098
  Total Number of Control Packets Received : 9096
  Total Number of Control Packets Dropped : 16

  Session VLAN  Source IP   Destination IP  Echo  State  Application
  -----------------  -----------  ----------------  -------  --------  ------------
  1    10  172.16.4.1  172.16.4.2  Enabled  Up  STATIC

switch(config)#

Switch 2:

switch(config)# ip route 15.212.178.0/24 172.16.4.1 bfd source-ip 172.16.4.2
switch(config)#
switch(config)# show run

Running configuration:

; J9850A Configuration Editor; Created on release #KB.16.04.0000x
; Ver #10:1b.7f.bb.ff.7c.59.fc.7b.ff.ff.fc.ff.ff.3f.ef:01
hostname "switch"
module A type j9989a
ip route 15.212.178.0 255.255.255.0 172.16.4.1 bfd source-ip 172.16.4.2
ip routing
snmp-server community "public" unrestricted
  oobm
  ip address dhcp-bootp
  exit
bfd enable
Bidirectional Forwarding Detection (BFD) Information

Administrative Status: Enabled
Echo Source IP: 3.3.3.3

Maximum number of sessions supported: 128
Number of sessions reserved for internal use: 0
Total Number of Sessions: 1
Number of Sessions Up: 1
Number of Sessions Down: 0

Global Statistics:
Total Number of Control Packets Transmitted: 9138
Total Number of Control Packets Received: 9119
Total Number of Control Packets Dropped: 27

<table>
<thead>
<tr>
<th>Session</th>
<th>VLAN</th>
<th>Source IP</th>
<th>Destination IP</th>
<th>Echo</th>
<th>State</th>
<th>Application</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>10</td>
<td>172.16.4.2</td>
<td>172.16.4.1</td>
<td>Enabled</td>
<td>Up</td>
<td>STATIC</td>
</tr>
</tbody>
</table>

Example

Configuring static route BFD using destination VLAN and BFD destination IP. All other BFD parameters (like detect multiplier, transmit/receive intervals, authentication) are automatically obtained from the associated VLAN interface of next-hop vlan-id. In the following example, the BFD parameters are obtained directly from VLAN 10.

Switch 1:

ip route <destination-network> <network-mask> vlan <gateway-vlan> bfd destination-ip <IP-Address-configured-on-peer-device>

BFD is enabled between self-device and the peer-device for the static route configured with gateway as Nexthop-Vlan-Id (Method 2):

switch(config)# ip route 192.172.4.0/24 vlan 10 bfd destination-ip 172.16.4.2
switch(config)#
switch(config)# show running-config

Running configuration:

; J9850A Configuration Editor; Created on release #KB.16.04.0000x
; Ver #10:1b.7f.bf.bb.ff.7c.59.fc.7b.ff.ff.ff.ff.3f.ef:01
hostname "switch"
module A type j9992a
ip route 192.172.4.0 255.255.255.0 vlan 10 bfd destination-ip 172.16.4.2
ip routing
snmp-server community "public" unrestricted

no allow-v2-modules

switch(config)# show bfd

Bidirectional Forwarding Detection (BFD) Information

Administrative Status : Enabled
Echo Source IP : 2.2.2.2

Maximum number of sessions supported : 128
Number of sessions reserved for internal use : 0
Total Number of Sessions : 1
Number of Sessions Up : 1
Number of Sessions Down : 0

Global Statistics:
Total Number of Control Packets Transmitted : 9226
Total Number of Control Packets Received : 9172
Total Number of Control Packets Dropped : 59

<table>
<thead>
<tr>
<th>Session</th>
<th>VLAN</th>
<th>Source IP</th>
<th>Destination IP</th>
<th>Echo</th>
<th>State</th>
<th>Application</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>10</td>
<td>172.16.4.1</td>
<td>172.16.4.2</td>
<td>Enabled</td>
<td>Up</td>
<td>STATIC</td>
</tr>
</tbody>
</table>

Switch 2:

switch(config)# ip route 0.0.0.0 0.0.0.0 172.16.4.1 bfd source-ip 172.16.4.2
switch(config)# show running-config

Running configuration:

; J9850A Configuration Editor; Created on release #KB.16.04.0000x
; Ver #10:1b.7f.bf.fb.7c.59.fc.7b.fb.fc.fc.fc.fc.fc.fc.fc.3f.ef:01
hostname "switch"
module A type j9989a
ip route 0.0.0.0 0.0.0.0 172.16.4.1 bfd source-ip 172.16.4.2
ip routing
snmp-server community "public" unrestricted

no allow-v2-modules

switch(config)# show bfd

Bidirectional Forwarding Detection (BFD) Information

Administrative Status : Enabled
Echo Source IP : 2.2.2.2

Maximum number of sessions supported : 128
Number of sessions reserved for internal use : 0
Total Number of Sessions : 1
Number of Sessions Up : 1
Number of Sessions Down : 0

Global Statistics:
Total Number of Control Packets Transmitted : 9226
Total Number of Control Packets Received : 9172
Total Number of Control Packets Dropped : 59

<table>
<thead>
<tr>
<th>Session</th>
<th>VLAN</th>
<th>Source IP</th>
<th>Destination IP</th>
<th>Echo</th>
<th>State</th>
<th>Application</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>10</td>
<td>172.16.4.1</td>
<td>172.16.4.2</td>
<td>Enabled</td>
<td>Up</td>
<td>STATIC</td>
</tr>
</tbody>
</table>

Switch 2:
switch(config)# show bfd-session

Bidirectional Forwarding Detection (BFD) Information

Administrative Status : Enabled
Echo Source IP 3.3.3.3

Maximum number of sessions supported : 128
Number of sessions reserved for internal use : 0
Total Number of Sessions : 1
Number of Sessions Up : 1
Number of Sessions Down : 0

Global Statistics:
Total Number of Control Packets Transmitted : 9256
Total Number of Control Packets Received : 9194
Total Number of Control Packets Dropped : 57

<table>
<thead>
<tr>
<th>Session</th>
<th>VLAN</th>
<th>Source IP</th>
<th>Destination IP</th>
<th>Echo</th>
<th>State</th>
<th>Application</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>10</td>
<td>172.16.4.2</td>
<td>172.16.4.1</td>
<td>Enabled</td>
<td>Up</td>
<td>STATIC</td>
</tr>
</tbody>
</table>

**Example**

Sharing a BFD session between STATIC and OSPF applications. When BFD is enabled for both STATIC and OSPF applications over the same pair of source and destination IPs, a single BFD session can be used to monitor the connectivity.

**Switch 1:**

switch(config)# ip route 200.1.12.0 255.255.255.0 172.16.4.2 bfd source-ip 172.16.4.1
switch(config)# show run

Running configuration:

; J9850A Configuration Editor; Created on release #KB.16.04.0000x
; Ver #10:1b.7f.bf.bb.ff.7c.59.fc.7b.ff.ff.fc.ff.ff.3f.ef:01
hostname "switch"
module A type j9992a
ip route 200.1.12.0 255.255.255.0 172.16.4.2 bfd source-ip 172.16.4.1
ip routing
snmp-server community "public" unrestricted
oobm
    ip address dhcp-bootp
    exit
router ospf
    area backbone
enable
exit
bfd enable
bfd echo-src-ip-address 2.2.2.2
vlan 1
   name "DEFAULT_VLAN"
   no untagged A1
   untagged A2-A21
   ip address dhcp-bootp
exit
vlan 10
   name "VLAN10"
   untagged A1
   bfd min-echo-receive-interval 700
   ip address 172.16.4.1 255.255.255.0
   ip ospf 172.16.4.1 area backbone
   ip ospf 172.16.4.1 bfd
exit
no allow-v2-modules

switch(config)# show bfd

Bidirectional Forwarding Detection (BFD) Information

   Administrative Status : Enabled
   Echo Source IP : 2.2.2.2

   Maximum number of sessions supported : 128
   Number of sessions reserved for internal use : 0
   Total Number of Sessions : 1
   Number of Sessions Up : 1
   Number of Sessions Down : 0

Global Statistics:
   Total Number of Control Packets Transmitted : 4330
   Total Number of Control Packets Received : 4333
   Total Number of Control Packets Dropped : 4

   +---+--------+-------------------+--------+----------+-----------------------------+
   | Session | VLAN | Source IP | Destination IP | Echo | State | Application |
   +---+--------+-------------------+--------+----------+-----------------------------+
   | 1 | 10 | 172.16.4.1 | 172.16.4.2 | Enabled | Up | OSPF/STATIC |

Switch 2:

switch(config)# ip route 0.0.0.0/0 172.16.4.1 bfd source-ip 172.16.4.2
switch(config)#
switch(config)# show running-config

Running configuration:

; J9850A Configuration Editor; Created on release #KB.16.04.0000x
; Ver #10:1b.7f.ff.7c.59.fc.7b.fc.ff.fc.ff.fc.ff.fc.ff.fc.3f.ef:01
hostname "switch"
module A type j9989a
ip route 0.0.0.0 0.0.0.0 172.16.4.1 bfd source-ip 172.16.4.2
ip routing
snmp-server community "public" unrestricted
oomb
   ip address dhcp-bootp
exit
router ospf
   area backbone
enable
exit
bfd enable
bfd echo-src-ip-address 3.3.3.3
vlan 1
    name "DEFAULT_VLAN"
    no untagged A1
    untagged A2-A24
    ip address dhcp-bootp
exit
vlan 10
    name "VLAN10"
    untagged A1
    bfd min-echo-receive-interval 700
    ip address 172.16.4.2 255.255.255.0
    ip ospf 172.16.4.2 area backbone
    ip ospf 172.16.4.2 bfd
exit
no allow-v2-modules
switch(config)# show bfd

Bidirectional Forwarding Detection (BFD) Information

  Administrative Status : Enabled
  Echo Source IP : 3.3.3.3

  Maximum number of sessions supported : 128
  Number of sessions reserved for internal use : 0
  Total Number of Sessions : 1
  Number of Sessions Up : 1
  Number of Sessions Down : 0

  Global Statistics:
  Total Number of Control Packets Transmitted : 8980
  Total Number of Control Packets Received : 8974
  Total Number of Control Packets Dropped : 0

<table>
<thead>
<tr>
<th>Session</th>
<th>VLAN</th>
<th>Source IP</th>
<th>Destination IP</th>
<th>Echo</th>
<th>State</th>
<th>Application</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
<td>172.16.4.2</td>
<td>172.16.4.1</td>
<td>Enabled</td>
<td>Up</td>
<td>OSPF/STATIC</td>
</tr>
</tbody>
</table>

Switch(config)#

Note with respect to the previous example:

- If BFD configuration is removed on SWITCH-1 on vlan-10 on EITHER OSPF or STATIC, still the BFD session is maintained as UP on SWITCH2 with OSPF/STATIC applications. SWITCH1 maintains the session only with BFD enabled application in this case.

- If OSPF or STATIC Configuration itself is removed on SWITCH-1 on vlan-10, still the BFD session is maintained as UP on SWITCH2 with OSPF/STATIC applications. SWITCH1 maintains the session only with configured application in this case.

- If BFD configuration is removed on SWITCH-1 on vlan-10 on BOTH OSPF and STATIC, the BFD session will go DOWN on SWITCH2 with OSPF/STATIC applications. SWITCH1 does not maintain any session.

- If OSPF and STATIC Configuration are removed on SWITCH-1 on vlan-10, the BFD session is maintained as DOWN on SWITCH2 with STATIC applications. SWITCH1 does not maintain any session.

- If OSPF adjacency breaks due to any reason (for example change in OSPF authentication etc) on SWITCH-1, BFD session will be maintained in UP state since STATIC is also registered on it.

Example
Sharing a single static route BFD session with multiple static routes having the same next-hop IP and BFD source IP.

**Switch 1:**

```plaintext
switch(config)# ip route 192.172.4.0/24 172.16.4.2 bfd source-ip 172.16.4.1
switch(config)# ip route 192.172.5.0/24 vlan 10 bfd destination-ip 172.16.4.2
switch(config)# ip route 192.172.6.0/24 172.16.4.2 bfd source-ip 172.16.4.1
```

```plaintext
switch(config)# show run

Running configuration:

; J9851A Configuration Editor; Created on release #KB.16.04.0000x
; Ver #10:1b.7f.bf.bb.ff.7c.59.fc.7b.ff.ff.fc.ff.ff.fc.ff.3f.ef:01
hostname "switch"
module A type j9992a
module L type j9987a
ip route 192.172.4.0 255.255.255.0 172.16.4.2 bfd source-ip 172.16.4.1
ip route 192.172.5.0/24 vlan 10 bfd destination-ip 172.16.4.2
ip route 192.172.6.0/24 172.16.4.2 bfd source-ip 172.16.4.1
ip routing
snmp-server community "public" unrestricted
oobm
   ip address dhcp-bootp
   exit
bfd enable
bfd echo-src-ip-address 2.2.2.2
vlan 1
   name "DEFAULT_VLAN"
   no untagged A1
   untagged A2-A21,L1-L24
   ip address dhcp-bootp
   exit
vlan 10
   name "VLAN10"
   untagged A1
   ip address 172.16.4.1 255.255.255.0
   exit
no allow-v2-modules
```

```plaintext
switch(config)# show bfd

Bidirectional Forwarding Detection (BFD) Information

  Administrative Status : Enabled
  Echo Source IP : 2.2.2.2

  Maximum number of sessions supported : 128
  Number of sessions reserved for internal use : 0
  Total Number of Sessions : 1
  Number of Sessions Up : 1
  Number of Sessions Down : 0

  Global Statistics:
  Total Number of Control Packets Transmitted : 158
  Total Number of Control Packets Received : 153
  Total Number of Control Packets Dropped : 203

  Session    VLAN  Source IP  Destination IP  Echo  State  Application
   -------  -----  ------------  -----------  -----  ------  -----------
   1        10    172.16.4.1  172.16.4.2  Enabled  Up    STATICVV
```
Switch 2:

```
switch(config)# ip route 15.212.178.0/24 172.16.4.1 bfd source-ip 172.16.4.2
switch(config)# show run

Running configuration:

; J9850A Configuration Editor; Created on release #KB.16.04.0000x
; Ver #10;lb.7f.ff.bb.ff.7c.59.fc.7b.ff.fc.ff.fc.7f.ef:01
hostname "switch"
module A type j9989a
ip route 15.212.178.0 255.255.255.0 172.16.4.1 bfd source-ip 172.16.4.2
ip routing
snmp-server community "public" unrestricted
oobm
    ip address dhcp-bootp
    exit
bfd enable
bfd echo-src-ip-address 3.3.3.3
vlan 1
    name "DEFAULT_VLAN"
    no untagged A1
    untagged A2-A24
    ip address dhcp-bootp
    exit
vlan 10
    name "VLAN10"
    untagged A1
    ip address 172.16.4.2 255.255.255.0
    exit
no allow-v2-modules

switch(config)# show bfd

Bidirectional Forwarding Detection (BFD) Information

  Administrative Status : Enabled
  Echo Source IP : 3.3.3.3

  Maximum number of sessions supported : 128
  Number of sessions reserved for internal use : 0
  Total Number of Sessions : 1
  Number of Sessions Up : 1
  Number of Sessions Down : 0

  Global Statistics:
  Total Number of Control Packets Transmitted : 9138
  Total Number of Control Packets Received : 9119
  Total Number of Control Packets Dropped : 27

<table>
<thead>
<tr>
<th>Session</th>
<th>VLAN</th>
<th>Source IP</th>
<th>Destination IP</th>
<th>Echo</th>
<th>State</th>
<th>Application</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>10</td>
<td>172.16.4.2</td>
<td>172.16.4.1</td>
<td>Enabled</td>
<td>Up</td>
<td>STATIC</td>
</tr>
</tbody>
</table>
```

A single static route BFD session can be shared with multiple static routes having same next-hop IP and BFD source IP. It is recommended that you keep the total number of static routes whose next-hop shares the same physical link to under 64.
show bfd

Syntax
show bfd-session [<session-id>]

Description
Shows Bidirectional Forwarding Detection (BFD) information. This command displays all the current BFD sessions in the switch. Detailed output is displayed if the user provides the session number.

Command context
config

Example
When a BFD session is shared by only static route application:

```bash
switch(config)# show bfd-session

Bidirectional Forwarding Detection (BFD) Information

Administrative Status : Enabled
Echo Source IP : 2.2.2.2

Maximum number of sessions supported : 128
Number of sessions reserved for internal use : 0
Total Number of Sessions : 1
Number of Sessions Up : 1
Number of Sessions Down : 0

Global Statistics:
Total Number of Control Packets Transmitted : 158
Total Number of Control Packets Received : 153
Total Number of Control Packets Dropped : 203

<table>
<thead>
<tr>
<th>Session</th>
<th>VLAN</th>
<th>Source IP</th>
<th>Destination IP</th>
<th>Echo</th>
<th>State</th>
<th>Application</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>10</td>
<td>172.16.4.1</td>
<td>172.16.4.2</td>
<td>Enabled</td>
<td>Up</td>
<td>STATIC</td>
</tr>
</tbody>
</table>
```

Example
When a BFD session is shared by both static route and an OSPF application:
switch(config)# show bfd-session

Bidirectional Forwarding Detection (BFD) Information

Administrative Status : Enabled
Echo Source IP : 2.2.2.2

Maximum number of sessions supported : 128
Number of sessions reserved for internal use : 0
Total Number of Sessions : 1
Number of Sessions Up : 1
Number of Sessions Down : 0

Global Statistics:
Total Number of Control Packets Transmitted : 16
Total Number of Control Packets Received : 15
Total Number of Control Packets Dropped : 4

<table>
<thead>
<tr>
<th>Session</th>
<th>VLAN</th>
<th>Source IP</th>
<th>Destination IP</th>
<th>Echo</th>
<th>State</th>
<th>Application</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>10</td>
<td>172.16.4.1</td>
<td>172.16.4.2</td>
<td>Enabled</td>
<td>Up</td>
<td>OSPF/STATIC</td>
</tr>
</tbody>
</table>

switch(config)# show bfd-session 1

BFD Session Information

Min Tx (sec) : 3
Min Rx (sec) : 3
Min Echo Rx (msec) : 500
Detect Multiplier : 5
Auth Mode : NONE
Password : ""
Application ID : OSPF/STATIC
Local Discriminator : 1
Remote Discriminator : 1
Echo : Enabled
Local Diag : No diagnostics configured.

<table>
<thead>
<tr>
<th>VLAN</th>
<th>Source IP</th>
<th>Destination IP</th>
<th>State</th>
<th>Pkt In</th>
<th>Pkt Drop</th>
<th>Pkt Out</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>172.16.4.1</td>
<td>172.16.4.2</td>
<td>Up</td>
<td>18</td>
<td>0</td>
<td>19</td>
</tr>
</tbody>
</table>

Smart link configuration commands

Set BFD source IP address for echo packets

This command helps to set a source IP address of echo packet for all BFD instances on the switch.

Syntax

```
bfd echo-src-ip-address IP-ADDR
no bfd echo-src-ip-address IP-ADDR
```

Description

Set the source IP address for BFD echo packets transmitted by the switch. The special value 0.0.0.0 will close only BFD echo sessions. Any existing control sessions will not be affected.

Options

```
echo-src-ip-address
```

Set the source IP address for BFD echo packets transmitted by the switch.
The source IP address for BFD echo packets transmitted by the switch.

**Set BFD source IP address for echo packets configuration**

```
switch(config)# bfd echo-src-ip-address 2.2.2.2
```

ATTENTION!! Make sure that echo packet source IP configured, does not belong to same subnet as the IP address assigned to any VLAN on the switch.

Running configuration:

```
; J9850A Configuration Editor; Created on release #KB.16.02.0000x
; Ver #0b:fc.59.f4.7b.ff.fc.ff.ff.3f.ef:0d
hostname "switch-name"
module A type j9899a
module F type j9987a
no rest-interface
password minimum-length 8
ip routing
snmp-server community "public" unrestricted
  oobm
    ip address dhcp-bootp
    exit
bfd enable
bfd echo-src-ip-address 2.2.2.2
vlan 1
  name "DEFAULT_VLAN"
  untagged A1-A24,F1-F24
  ip address dhcp-bootp
  exit
no allow-v2-modules
```

**Show BFD-session**

```
switch# show bfd-session
```

Bidirectional Forwarding Detection (BFD) Information

```
   Admin Status : Enabled
   Echo source IP : 2.2.2.2

   Session VLAN Source IP Destination IP Echo State application
  ===== ===== ========= ========= ============= ===========
        ===== ===== ========= ========= ============= ===========

switch#
```

```
switch# show bfd-session 1
```

BFD Session Information
BFD session 1 not found

**Enable BFD session**

**Syntax**

```
bfd enable| disable echo-src-ip-address
```

**Description**

Configure Bidirectional Forwarding Detection (BFD) global settings.

**options**

```
enable
```

Enable BFD globally.
disable
Disable BFD globally.

echo-src-ip-address
Set the source IP address for BFD echo packets transmitted by the switch.

Enable BFD session: Configuration

; J9850A Configuration Editor; Created on release #KB.16.02.0000x
; Ver #0b:fc.59.f4.7b.ff.ff.fc.ff.3f.ef:0d
hostname "switch"
module A type j9989a
module F type j9987a
no rest-interface
password minimum-length 8
ip routing
snmp-server community "public" unrestricted
oobm
  ip address dhcp-bootp
exit
bfd enable
vlan 1
  name "DEFAULT_VLAN"
  untagged A1-A24,F1-F24
  ip address dhcp-bootp
exit
no allow-v2-modules

switch# show bfd-session

Bidirectional Forwarding Detection (BFD) Information

  Admin Status : Enabled
  Echo source IP :

  Session   VLAN   Source IP  Destination IP  Echo    State  Application

switch# show bfd-session 1

BFD Session Information
BFD session 1 not found

Enable debug logging
This command enables logging BFD packet fields and/or BFD events for the identified session.

Syntax
Parameters debug bfd packet | event session SESS-ID

Description
Enable BFD debug logging.

Options
BFD
Enable BFD debug logging.

session
Display debug messages for a specific session.

SESS-ID
A BFD session number to display debug messages for.

**packet**
Display important fields of BFD packets.

**event**
Display BFD state machine events.

### Clear BFD statistics

This command helps to reset the specified BFD counter.

**Syntax**
```
clear statistics bfd SESS-ID
```

**Description**
Reset Bidirectional Forwarding Detection (BFD) statistics.

**Options**

**BFD**
Reset Bidirectional Forwarding Detection (BFD) statistics.

**SESS-ID**
The id of the session whose statistics should be cleared.

### Clear BFD statistics: Before clearing statistics

```
switch# sh bfd

Bidirectional Forwarding Detection (BFD) Information

    Admin Status : Enabled
    Echo source IP : 2.2.2.2

    Global Statistics:
    Total Number of Control Packets Transmitted : 13
    Total Number of Control Packets Received : 13
    Total Number of Control Packets Dropped : 0

    VLAN  SourceIP  DestIP     Echo   State   Application
    ------  ---------  -------  -----  ------  -------------
          100.100.100.100  100.100.100.101   Enabled   Up

switch# show bfd-session 1

BFD Session Information - Session 1

    Min Tx Interval (sec) : 10
    Min Rx Interval (sec) : 10
    Min Echo Rx Interval (msec) : 700
    Detect Multiplier : 3
    Authentication Mode : NONE
    Password : ""
    Application : OSPF
    Local Discriminator : 1
    Remote Discriminator : 1
    Echo : Enabled
    Local Diagnostic : No diagnostics configured.

    VLAN  Source IP  Destination IP  State  Pkt In  Pkt Drop  Pkt Out
    ----  ----------  ---------------  -----  ------  ------  ------
          100.100.100.100  100.100.100.101  Up  322    0    320
```
Clear BFD statistics: After clearing BFD statistics

switch# sh bfd

Bidirectional Forwarding Detection (BFD) Information

Admin Status : Enabled
Echo source IP : 2.2.2.2

Global Statistics:
Total Number of Control Packets Transmitted : 0
Total Number of Control Packets Received : 0
Total Number of Control Packets Dropped : 0

<table>
<thead>
<tr>
<th>Session</th>
<th>VLAN</th>
<th>Source IP</th>
<th>Dest IP</th>
<th>Echo</th>
<th>State</th>
<th>Application</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>20</td>
<td>100.100.100.100</td>
<td>100.100.100.101</td>
<td>Enabled</td>
<td>Up</td>
<td>OSPF</td>
</tr>
</tbody>
</table>

switch# show bfd-session 1

BFD Session Information - Session 1

Min Tx Interval (sec) : 10
Min Rx Interval (sec) : 10
Min Echo Rx Interval (msec) : 700
Detect Multiplier : 3
Authentication Mode : NONE
Password : ""
Application : OSPF
Local Discriminator : 1
Remote Discriminator : 1
Echo : Enabled
Local Diagnostic : No diagnostics configured.

<table>
<thead>
<tr>
<th>VLAN</th>
<th>Source IP</th>
<th>Destination IP</th>
<th>State Pkt In</th>
<th>Pkt Drop</th>
<th>Pkt Out</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>100.100.100.100</td>
<td>100.100.100.100</td>
<td>Up</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

VXLAN show commands

VXLAN show commands include commands to display the status of a VXLAN feature, tunnels, and tunnel statistics.

Show all BFD sessions

This command displays all the current BFD sessions in the switch. Detailed output is displayed if the user provides the session number.

Auth Mode information is shown in the command bfd-session SESSION-NUMBER.

Syntax

show bfd-session

Description

Show Bidirectional Forwarding Detection (BFD) information.

Show all BFD sessions
Show the details of a particular BFD session

This command displays the details of a particular BFD session on the switch. User can obtain VLAN to session ID mapping through show BFD session command.

**Syntax**

```
show bfd-session SESS-ID
```

This command will display the detailed output for the session number provided in the argument.

**Show particular BFD session**

```
Switch# show bfd-session 1

BFD Session Information - Session 1

Min Tx (in secs) : 3
Min Rx (in secs) : 3
Min Echo Rx (in msecs) : 500
Detect multiplier : 5
Auth Mode : keyed-SHA1
Password : "hp123"
Application : OSPF
Local Discriminator : 1
Remote Discriminator : 1
Echo : Enabled
Local Diagnostic : No diagnostics configured.

<table>
<thead>
<tr>
<th>VLAN Source IP</th>
<th>Destination IP</th>
<th>State</th>
<th>Pkt In</th>
<th>Pkt Drop</th>
<th>Pkt Out</th>
</tr>
</thead>
<tbody>
<tr>
<td>3 100.100.100.100 100.100.100.101</td>
<td>Up</td>
<td>322</td>
<td>0</td>
<td>320</td>
<td></td>
</tr>
</tbody>
</table>

```

HP-Switch-5412Rz12# show bfd 1

BFD Session Information

Min Echo Rx (in msecs) : 700

<table>
<thead>
<tr>
<th>Session VLAN Source IP</th>
<th>Destination IP</th>
<th>Echo</th>
<th>State</th>
<th>Application</th>
</tr>
</thead>
<tbody>
<tr>
<td>2 10 100.100.100.100 100.100.100.101</td>
<td>Enabled  Up</td>
<td>VRRP</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

```
**Prerequisites**

- Before enabling BFD globally, disable Internet Control Message Protocol (ICMP) redirect messages using the `Parameters ip icmp redirect` command. Additionally, this has to be configured on the VRRP main router if the backup has a BFD session monitoring the main router.
- Ensure that virus throttling is disabled before enabling BFD globally.
- Ensure that BFD timers are configured as per deployment needs. Configuring the timers to be too aggressive (for example, detect-multiplier of 1) can sometimes lead to BFD session flaps depending upon traffic conditions.
- BFD works only for directly connected neighbors. BFD neighbors must be no more than one IP hop away. Multihop configurations are not supported.
- Demand mode is not supported.
- BFD over IPv6 is not supported.
- Recommended multiplier value should be minimum 3 when min-echo-receive interval or transmit intervals are aggressive.
- Only BFD version 1 is supported.
- When BFD sessions are using authentication, the sessions might go down after a switchover.
- Priority settings for BFD packets: When the interfaces are over-subscribed, BFD packets have to be prioritized, so that these pkts are not overwhelmed by other data packets. CoS values must be explicitly configured in the appropriate egress QoS service policy. CoS values for BFD packets can be set using the qos command. For example, ‘qos udp-port ipv4 3785 priority 6’ assigns 802.1p priority 6 to BFD echo packets. Mapping of 802.1p priorities to egress queues can be found using the show qos queue-config command. These CoS settings must be applied on all intermediate switches if any so that BFD packets get prioritized all through the way.
- VRRP Advertise interval has to be configured with higher values (ex- 10) to avoid the BFD session flaps due to VRRP packet loss.

---

**WARNING**

Do not clear IP host table using the `clear arp` command on a system with active BFD sessions, because this can lead to BFD sessions flapping.
Chapter 19
Support and other resources

Accessing Aruba Support

<table>
<thead>
<tr>
<th>Support Services</th>
<th><a href="https://www.arubanetworks.com/support-services/">https://www.arubanetworks.com/support-services/</a></th>
</tr>
</thead>
<tbody>
<tr>
<td>North America telephone</td>
<td>1-800-943-4526 (US &amp; Canada Toll-Free Number)</td>
</tr>
<tr>
<td></td>
<td>+1-408-754-1200 (Primary - Toll Number)</td>
</tr>
<tr>
<td></td>
<td>+1-650-385-6582 (Backup - Toll Number - Use only when all other numbers are not working)</td>
</tr>
<tr>
<td>International telephone</td>
<td><a href="https://www.arubanetworks.com/support-services/contactsupport/">https://www.arubanetworks.com/support-services/contactsupport/</a></td>
</tr>
</tbody>
</table>

Be sure to collect the following information before contacting Support:

- Technical support registration number (if applicable)
- Product name, model or version, and serial number
- Operating system name and version
- Firmware version
- Error messages
- Product-specific reports and logs
- Add-on products or components
- Third-party products or components

Other useful sites

Other websites that can be used to find information:

<table>
<thead>
<tr>
<th>Airheads social forums and Knowledge Base</th>
<th><a href="https://community.arubanetworks.com/">https://community.arubanetworks.com/</a></th>
</tr>
</thead>
<tbody>
<tr>
<td>Software licensing</td>
<td><a href="https://lms.arubanetworks.com/">https://lms.arubanetworks.com/</a></td>
</tr>
<tr>
<td>End-of-Life information</td>
<td><a href="https://www.arubanetworks.com/support-services/end-of-life/">https://www.arubanetworks.com/support-services/end-of-life/</a></td>
</tr>
<tr>
<td>Aruba software and documentation</td>
<td><a href="https://asp.arubanetworks.com/downloads">https://asp.arubanetworks.com/downloads</a></td>
</tr>
</tbody>
</table>

Accessing updates

To download product updates:

Aruba Support Portal
If you are unable to find your product in the Aruba Support Portal, you may need to search My Networking, where older networking products can be found:

**My Networking**

[https://www.hpe.com/networking/support](https://www.hpe.com/networking/support)

To view and update your entitlements, and to link your contracts and warranties with your profile, go to the Hewlett Packard Enterprise Support Center More Information on Access to Support Materials page: [https://support.hpe.com/portal/site/hpsc/aae/home/](https://support.hpe.com/portal/site/hpsc/aae/home/)

---

**NOTE**

Access to some updates might require product entitlement when accessed through the Hewlett Packard Enterprise Support Center. You must have an HPE Passport set up with relevant entitlements.

Some software products provide a mechanism for accessing software updates through the product interface. Review your product documentation to identify the recommended software update method.

To subscribe to eNewsletters and alerts: [https://www.hpe.com/support/e-updates](https://www.hpe.com/support/e-updates)

---

**Warranty information**


**Regulatory information**


**Additional regulatory information**

Aruba is committed to providing our customers with information about the chemical substances in our products as needed to comply with legal requirements, environmental data (company programs, product recycling, energy efficiency), and safety information and compliance data, (RoHS and WEEE). For more information, see [https://www.arubanetworks.com/company/about-us/environmental-citizenship/](https://www.arubanetworks.com/company/about-us/environmental-citizenship/).

**Documentation feedback**

Aruba is committed to providing documentation that meets your needs. To help us improve the documentation, send any errors, suggestions, or comments to Documentation Feedback (hpe-aruba-techpub-india@hpe.com). When submitting your feedback, include the document title, part number, edition, and publication date located on the front cover of the document. For online help content, include the product name, product version, help edition, and publication date located on the legal notices page.
Overview of Apple's Bonjour and Google's Chromecast

The Aruba mDNS Gateway and Google Chromecast solution adds support for Apple’s Bonjour and Google’s Chromecast discovery from a switch. The solution uses mDNS protocol for discovery and is responsible for handling mDNS packets.

Bonjour

Aruba’s mDNS Gateway solution supports Apple’s Bonjour protocol to the switch. Bonjour is Apple’s implementation of a suite of zero-configuration networking protocols and is supported by both Mac OS X devices (such as laptops and desktops), and Apple iOS devices (such as iPhones and iPads). Bonjour’s zero-configuration networking services benefits include:

- No longer having to assign IP addresses or host names to access network services on Mac OS X and Apple iOS devices
- Applications can leverage Bonjour to automatically detect required services.
- Interacts with other applications to allow for automatic connection of devices.
- Communication and data exchange is possible without user configuration.

Google’s Chromecast

Chromecast is a digital media player developed by Google. The device is a HDMI dongle that plays audio and video content on a high-definition screen by directly streaming it via Wi-Fi from the Internet or a local network. The media is selected, by users, to play on devices by enabling Chromecast mobile and web applications. Casting a tab for sites that are not Google Cast-enabled. mirrors most Google Chrome browser content running on the device (MAC OSX and Windows).

Chromecast uses a simple multicast protocol for discovery and launch. This protocol enables users to mirror their devices on a second screen.

Aruba mDNS protocol

Aruba supports mDNS protocol implemented as a server. mDNS is the primary method of discovering a Chromecast that supports the v2 API. While SSDP/DIAL support is still present and used by some applications (such as You Tube), existing applications have to migrate to the new SDK using the new protocol.

mDNS Gateway

The mDNS gateway, running on a switch, will listen for Bonjour responses and Bonjour queries and forward them to different subnets. Its main function is to forward Bonjour traffic by re-transmitting the traffic between reflection enabled VLANs. The switches are configured interfaces in the VLANs for which they are performing packet reflection.

Note

The mDNS gateway in a switch acts as an application layer gateway between subnets. An IP interface is required on each of the network that it is reflecting between.
Service filtering

The mDNS profiles feature is responsible for applying filter profiles to mDNS resource records in mDNS response/query packets. The mDNS response/query can be filtered to give better control of the services. Service filtering allows network administrators to manipulate both the responses sent to and coming from clients in order to allow or deny mDNS services. This mechanism prevents clients from being aware of both specified services and announce specific services. These filters can be outbound from the switch to clients or inbound from clients to the switch. Profiles can be applied per-VLAN.

There is a global default which allows or denies traffic that does not match any rule. After a match is found other filter rules are ignored.

---

**NOTE**
Service filtering cannot block the connection between devices. For example, if the client knows the remote device’s IP address, they can still establish a connection without utilizing the mDNS protocol. Service filtering functions to keep names and addresses out services out of mDNS responses.

---

**mDNS query and response assessment**

---

**Wireless printer service process**

Process overview of service for a wireless printer:

1. Wireless Printer 1 sends an mDNS response advertising printer services in Switch 1 on VLAN 1.
2. Switch 1 has no inbound filter in VLAN 1. The global filter set to permit all.
3. Switch 1 checks the outbound filter in VLAN 1. As there is no specific outbound filter, the global status is **permit all**. It will flood the packet in VLAN 1 except the source port.
4. iPhone 1 in VLAN 1 receives the service announcement.
5. Switch 1 checks the reflection status. Reflection is enabled on VLAN 2 and 3.
6. Switch 1 checks the outbound filter in VLAN 2. As there is no specific outbound filter, it will forward the service announcement in VLAN 2.
7. Default action **permit all**.
8. Switch 1 checks the outbound filter in VLAN 3. The outbound filter is set to **deny wireless printer** therefore the packet will not be forwarded to VLAN 3.
9. Switch 3 receives the service advertisement in VLAN 2. It will flood the packet in VLAN 2 except the source port.
10. Host 2 in Switch 3 receives the service announcement.

**Wireless Printer advertising printer service**

The following procedure depicts an advertising service process for a wireless printer in the form of an example.

1. Wireless Printer 2 sends an mDNS response advertising printer services in VLAN 3.
2. Switch 2 does not have any inbound filter in VLAN 3, so it receives the wireless printer service announcement.
3. Switch 2 checks the outbound filter in VLAN 3. There is no specific outbound filter on VLAN 3, so it floods the service announcement in VLAN 3 (except at the source port.)
4. Switch 2 checks the reflection status. Since switch 2 is not enabled, switch 2 does not forward.
5. As there is no inbound filter in VLAN 3 of switch 1, it receives the service announcement on VLAN 3. When switch 1 checks the outbound filter in VLAN 3, there is deny operation for service type wireless printer error message. Therefore switch 1 will not flood the packet in VLAN 3.
6. Switch 1 checks the reflection status. The reflection is enabled on VLAN 2 and 3 however VLAN 3 is incoming so the reflection will not function. In VLAN 2 it checks the outbound filter. There is no outbound filter in VLAN 2 so switch 1 forwards the service announcement in VLAN 2.
7. Switch 3 does not have any inbound filter therefore it receives service announcements in VLAN 2.
8. Switch 2 checks the outbound filter in VLAN 2. As there is no specific outbound filter, the global action is to permit all so switch 2 floods the packet in VLAN 2 (except the source port.)
9. Host 2 receives the switch 2 print service announcement.

**Host 2 queries for printers**

The following procedure depicts a service process for mDNS queries for a wireless printer in the form of an example.

1. Host 2 sends an mDNS query for printers.
2. There is no inbound filter in VLAN 2 of Switch 3 therefore it receives the query.
3. Switch 3 checks the outbound filter in VLAN 2. As there is no specific outbound filter the default action is permit all.
4. Switch 3 floods the query in VLAN 2 (except the source port.)
5. Switch 1 receives the query and check the inbound filters. Permit for the instance name, Host 2, allows the packet on VLAN 2.

6. Switch 1 checks the outbound filter for VLAN 2. As there is no specific filter and global filter is permit all, it will flood the packet in VLAN 2 (except the source port.)

7. Switch 1 checks the reflection status. Reflection is enabled on VLAN 2 and VLAN 3. Since VLAN 2 is an incoming VLAN, it will not pass the reflection on VLAN 2.

8. Switch 1 checks the outbound filters on VLAN 3. There is no rule to deny Host 2 query and the global filter is set to permit all so it will forward the packet to VLAN 3.

9. Switch 2 receives the service and checks for any inbound and outbound filters in VLAN 3.

10. There is no specific inbound and outbound filter in VLAN 3 therefore it will flood the query in VLAN 3 (except the source port.)

11. Reflection is not enabled in Switch 2 therefore it will not pass any further reflection.

12. Wireless printer 2 responses to the query and switch 2 does not have any inbound and outbound filters therefore it will flood the response to VLAN 3 (except the source port.)

13. Switch 1 receives the packet as there are no inbound filters in VLAN 3. VLAN 3 has an outbound filter set to deny wireless printer service. The service will not flood VLAN 3.

14. Switch 1 checks the reflection status which is enabled in VLAN 2 and 3. Since the incoming VLAN is 3, the packet will not forward to VLAN 3.

15. Switch 1 checks the outbound filter in VLAN 2. As there is no specific filter, it will forward the response to VLAN 2.

16. Switch 3 receives the response on VLAN 2 as there is no inbound filter to deny this service.

17. Switch 3 does not have any outbound filters in VLAN 2, so it will flood the response in VLAN 2 (except the source port.)


### iPhone 1 queries for printers

The following depicts a service process for iPhone queries for a wireless printer in the form of an example.

1. iPhone 1 sends an mDNS query for printers in switch 1 on VLAN 1.

2. Switch 1 checks the inbound filter in VLAN 1. As there is no specific filters, it receives the query.

3. Switch 1 checks the outbound filter in VLAN 1. As there is no specific filter therefore it flood the packet in VLAN 1 (except the source port.)

4. Switch 1 checks the reflection status. The reflection is enabled on VLAN 2 and 3.

5. Switch 1 checks the outbound filters on VLAN 2 and 3. In VLAN 3 the outbound filter is set to deny wireless printer therefore it will not reflect the packet to VLAN 3. There is no specific outbound filter in VLAN 2 so it will forward the packet to VLAN 2.

6. In switch 1, wireless printer 1 receives the iPhone 1 query and sends a response. Switch 1 checks the inbound filter, outbound filter and floods the response to VLAN 1 (except the source port.)

7. Switch 3 receives the iPhone 1 query and floods the packet in VLAN 2. As there is no specific inbound and outbound filters in switch 3, there is no associated printers in switch 3. There will not be any further response.
Limitations of the mDNS gateware and Chromecast

The following are limitations of the mDNS gateway and Chromecast features:

- IPv6 is not supported.
- In distributed environment enable gateway in one switch to avoid loops.
- Chromecast v1 (DIAL over SSDP) is not supported.
- Custom filters are not supported. For example:
  
  ```
  rule <name> service *tv*
  ```

  ```
  rule <name> instance *ipad*
  ```

- Instance name filter can only be used in the following format:
  ```
  <instance_name>.<service_name>.<domain_name>
  ```

- mDNS commands are not available from the web and the menu.
- If the user configures both permit and deny for same service-instance and assign that to same VLAN then it is not valid configuration. System will not behave properly.
- If the user has detected the Chromecast device via a permit profile VLAN and is doing a transition to deny profile, VLAN will need to clean the cache memory. Otherwise the system might get connected with already discovered device. It won't try to discover it again. This is an expected behavior.

Profile and rule limit details

The following are the profile and rule limit details for BTTF and SMB:

**BTTF:**
- Max Profile: 50
- Max rule per profile: 15
- Max vlan per profile: 10
- Max Profile: 25
- Max rule per profile: 15
- Max vlan per profile: 5

**SMB:**
- Max gateway VLAN is depends on Max VLAN supported on the switches.

Syntax

```
mdns enable
```

**Syntax**

```
mdns enable
```

**Description**

Enable or disables mDNS gateway support on switch.

The `no` form of the command disables mdns. The default value is disabled.

**Command context**

`config` with `manager` permissions.

**Create mDNS reflection**

This command is supported in the `config` context.

**Syntax**
mdns gateway vlan VLAN-LIST
mdns gateway vlan VLAN-LIST

**Description**

Configures the VLAN reflection for mDNS traffic. If the VLAN is not set, the mDNS traffic will not flood to different subnets, it will only flood to the incoming VLAN.

**Options**

**gateway**

Enable VLAN for mDNS gateway.

---

**Create or delete a mDNS profile**

This command will be supported on config context in manager mode. This is a context command. Separate context is created for this.

**Syntax**

```markdown
mdns profile PROFILE-NAME
no mdns profile PROFILE-NAME
```

**Description**

Create or delete an mDNS profile.

**rule**

**Syntax**

```markdown
rule rule-id instance | service NAME action permit | deny
no rule rule-id instance | service NAME action permit | deny
```

**Description**

Sets rules for each mDNS profile. You can configure specific rule to permit or deny the mDNS packet.

**Command context**

`config` This command is supported in the mDNS profile context.

**Parameters**

**rule**

Create or delete a rule for mDNS profile.

**instance**

Instance name of the client.

**service**

Service name of the client.

**action**

Specify the action for mDNS traffic.

**permit**

Permit the packet upon successful match.

**deny**

Deny the packet upon successful match.
Set the specific mDNS profile for VLAN

This command is supported in the mDNS profile context.

**Syntax**

```plaintext
vlan VLAN-LIST
no vlan VLAN-LIST
```

**Description**

Used to set the mDNS profile for a particular VLAN. Based on the rule, the filter permits or denies traffic.

**Options**

VLAN-LIST

Set the global mDNS profile

This command is supported in the configure context in manager mode.

**Syntax**

```plaintext
mdns default filter in | out action permit | deny
```

**Description**

Used to set the default action for all VLANs. If there is no specific rule for a particular VLAN, the default action will be applied. By default, the global action is set to deny for both inbound and outbound traffic.

**Options**

- **filter**
  - **in**: Match inbound traffic.
  - **out**: Match outbound traffic.
- **default**: Set the action of the mDNS default filter

Show mdns

**Syntax**

```plaintext
show mdns
```

**Description**

Display the status of the mDNS feature.

**Command context**

`config`

**Parameters**

- **mDNS**: Display the status of the mDNS feature

**Examples**
Show mDNS gateway

Syntax
show mdns gateway

Description
Display the reflection VLAN list of the mDNS gateway.

Options
gateway
mDNS gateway

Example

show mdns gateway

mDNS Gateway Configuration
Gateway VLAN List: 1-10,12

Show mDNS profile configuration

Syntax
show mdns profile

Description
Display mDNS profile configuration information.

Options
profile
mDNS profile information

Example

mDNS profile configuration
Profile Name: Students
VLANs : 1-3,25

Rules:
ID  Instance  Service   Action
---  !!!!!!    !!!!!    -------
1    ANY      AppleTV  Deny
2    MyComputer  ANY  Permit

Profile Name: Professors
VLANs : 3-6,10

Rules:
ID  Instance  Service   Action
---  !!!!!!    !!!!!    -------
1    ANY      AppleTV  Deny
2    MyComputer  ANY  Permit
Show mDNS profile name

Syntax

```
show mdns profile PROFILE-NAME
```

Description

Display mDNS profile name information.

Options

PROFILE-NAME

Specify the profile name.

Example

```
mDNS profile configuration
    Profile Name: Students
    VLANs : 1-3,25

    Rules:
    ID  Instance   Service    Action
    ---  ---------  ----------  -------
    1    ANY       AppleTV    Deny
    2    MyComputer ANY    Permit
```

Show mDNS

```
mDNS enable
mDNS gateway vlan 1-2
mDNS profile "abcd"
rule 1 instance Host1 action permit
rule 2 service AppleTv action deny
vlan 1-2
exit

vlan 1
    name "DEFAULT_VLAN"
    untagged 1-24
    ip address dhcp-bootp
    exit

vlan 2
    name "VLAN2"
    untagged 2
    ip address 10.1.1.1 255.255.255.0
    exit
```

Debug mDNS

Syntax

```
debug mdns
```

Description

Enable or disable mDNS debug logging.

Usage

```
debug mdns
no debug mdns
```