JUNOS® Software

Routing Protocols Configuration Guide

Release 10.0

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

Published: 2009-10-13

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

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

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

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

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

Juniper Networks, the Juniper Networks logo, JUNOS, NetScreen, ScreenOS, and Steel-Belted Radius are registered trademarks of Juniper Networks, Inc. in the United States and other countries. JUNOSe is a trademark of Juniper Networks, Inc. All other trademarks, service marks, registered trademarks, or registered service marks are the property of their respective owners.

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

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

JUNOS® Software Routing Protocols Configuration Guide,
Release 10.0
Copyright © 2009, Juniper Networks, Inc.
All rights reserved. Printed in USA.

Writing: Ines Salazar
Editing: Nancy Kurahashi
Illustration: Faith Bradford, Nathaniel Woodward
Cover Design: Edmonds Design

Revision History
October 2009—JUNOS 10.0R1

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

YEAR 2000 NOTICE

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

END USER LICENSE AGREEMENT

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

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

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

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

a. Customer shall use Embedded Software solely as embedded in, and for execution on, Juniper equipment originally purchased by Customer from Juniper or an authorized Juniper reseller.

b. Customer shall use the Software on a single hardware chassis having a single processing unit, or as many chassis or processing units for which Customer has paid the applicable license fees; provided, however, with respect to the Steel-Belted Radius or Odyssey Access Client software only, Customer shall use such Software on a single computer containing a single physical random access memory space and containing any number of processors. Use of the Steel-Belted Radius or IMS AAA software on multiple computers or virtual machines (e.g., Solaris zones) requires multiple licenses, regardless of whether such computers or virtualizations are physically contained on a single chassis.

c. Product purchase documents, paper or electronic user documentation, and/or the particular licenses purchased by Customer may specify limits to Customer's use of the Software. Such limits may restrict use to a maximum number of seats, registered endpoints, concurrent users, sessions, calls, connections, subscribers, clusters, nodes, realms, devices, links, ports or transactions, or require the purchase of separate licenses to use particular features, functionalities, services, applications, operations, or capabilities, or provide throughput, performance, configuration, bandwidth, interface, processing, temporal, or geographical limits. In addition, such limits may restrict the use of the Software to managing certain kinds of networks or require the Software to be used only in conjunction with other specific Software. Customer's use of the Software shall be subject to all such limitations and purchase of all applicable licenses.

d. For any trial copy of the Software, Customer's right to use the Software expires 30 days after download, installation or use of the Software. Customer may operate the Software after the 30-day trial period only if Customer pays for a license to do so. Customer may not extend or create an additional trial period by re-installing the Software after the 30-day trial period.

e. The Global Enterprise Edition of the Steel-Belted Radius software may be used by Customer only to manage access to Customer's enterprise network. Specifically, service provider customers are expressly prohibited from using the Global Enterprise Edition of the Steel-Belted Radius software to support any commercial network access services.

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

4. **Use Prohibitions.** Notwithstanding the foregoing, the license provided herein does not permit the Customer to, and Customer agrees not to and shall not: (a) modify, unbundle, reverse engineer, or create derivative works based on the Software; (b) make unauthorized copies of the Software (except as necessary for backup purposes); (c) rent, sell, transfer, or grant any rights in and to any copy of the Software, in any form, to any third party; (d) remove any proprietary notices, labels, or marks on or in any copy of the Software or any product in which the Software is embedded; (e) distribute any copy of the Software to any third party, including as may be embedded in Juniper equipment sold in the secondhand market; (f) use any 'locked' or key-restricted feature, function, service, application, operation, or capability without first purchasing the applicable license(s) and obtaining a valid key from Juniper, even if such feature, function, service, application, operation, or capability is enabled without a key; (g) distribute any key for the Software provided by Juniper to any third party; (h) use the Software in any manner that extends or is broader than the uses purchased by Customer from Juniper or an authorized Juniper reseller; (i) use Embedded Software on non-Juniper equipment; (j) use Embedded Software (or make it available for use) on Juniper equipment that the Customer did not originally purchase from Juniper or an authorized Juniper reseller; (k) disclose the results of testing or benchmarking of the Software to any third party without the prior written consent of Juniper; or (l) use the Software in any manner other than as expressly provided herein.

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

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

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

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

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

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

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

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

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

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

15. **Miscellaneous.** This Agreement shall be governed by the laws of the State of California without reference to its conflicts of laws principles. The provisions of the U.N. Convention for the International Sale of Goods shall not apply to this Agreement. For any disputes arising under this Agreement, the Parties hereby consent to the personal and exclusive jurisdiction of, and venue in, the state and federal courts within Santa Clara County, California. This Agreement constitutes the entire and sole agreement between Juniper and the Customer with respect to the Software, and supersedes all prior and contemporaneous

agreements relating to the Software, whether oral or written (including any inconsistent terms contained in a purchase order), except that the terms of a separate written agreement executed by an authorized Juniper representative and Customer shall govern to the extent such terms are inconsistent or conflict with terms contained herein. No modification to this Agreement nor any waiver of any rights hereunder shall be effective unless expressly assented to in writing by the party to be charged. If any portion of this Agreement is held invalid, the Parties agree that such invalidity shall not affect the validity of the remainder of this Agreement. This Agreement and associated documentation has been written in the English language, and the Parties agree that the English version will govern. (For Canada: Les parties aux présentés confirment leur volonté que cette convention de même que tous les documents y compris tout avis qui s'y rattaché, soient redigés en langue anglaise. (Translation: The parties confirm that this Agreement and all related documentation is and will be in the English language)).

Abbreviated Table of Contents

		About This Guide	xxxvii
Part 1		**Overview**	
	Chapter 1	Routing Protocols Concepts	3
	Chapter 2	Complete Routing and Routing Protocol Configuration Statements	15
Part 2		**Protocol-Independent Routing Properties**	
	Chapter 3	Protocol-Independent Routing Properties Overview	45
	Chapter 4	Configuring Routing Tables and Routes	51
	Chapter 5	Configuring Other Protocol-Independent Routing Properties	111
	Chapter 6	Configuring Logical Systems	135
	Chapter 7	Summary of Protocol-Independent Routing Properties Configuration Statements	141
Part 3		**Routing Instances**	
	Chapter 8	Introduction to Routing Instances	225
	Chapter 9	Routing Instances Configuration Guidelines	229
	Chapter 10	Summary of Routing Instances Configuration Statements	273
Part 4		**Multitopology Routing**	
	Chapter 11	Introduction to Multitopology Routing	287
	Chapter 12	Multitopology Routing Configuration Guidelines	291
	Chapter 13	Summary of Multitopology Routing Configuration Statements	303
Part 5		**Interior Gateway Protocols**	
	Chapter 14	Introduction to IS-IS	315
	Chapter 15	IS-IS Configuration Guidelines	321
	Chapter 16	Summary of IS-IS Configuration Statements	373
	Chapter 17	Introduction to OSPF	431
	Chapter 18	OSPF Configuration Guidelines	443
	Chapter 19	Summary of OSPF Configuration Statements	499
	Chapter 20	Introduction to RIP	581
	Chapter 21	RIP Configuration Guidelines	583

Chapter 22	Summary of RIP Configuration Statements	605
Chapter 23	Introduction to RIPng	631
Chapter 24	RIPng Configuration Guidelines	633
Chapter 25	Summary of RIPng Configuration Statements	643
Chapter 26	Introduction to ICMP Router Discovery	659
Chapter 27	ICMP Router Discovery Configuration Guidelines	661
Chapter 28	Summary of ICMP Router Discovery Configuration Statements	665
Chapter 29	Introduction to Neighbor Discovery	675
Chapter 30	Neighbor Discovery Configuration Guidelines	677
Chapter 31	Summary of Neighbor Discovery Router Advertisement Configuration Statements	685
Chapter 32	Secure Neighbor Discovery Configuration Guidelines	699
Chapter 33	Summary of Secure Neighbor Discovery Configuration Statements	703

Part 6 BGP

Chapter 34	Introduction to BGP	713
Chapter 35	BGP Configuration Guidelines	721
Chapter 36	Summary of BGP Configuration Statements	803

Part 7 Indexes

	Index	881
	Index of Statements and Commands	901

Table of Contents

About This Guide xxxvii

JUNOS Documentation and Release Notes xxxvii
Objectives xxxviii
Audience xxxviii
Supported Platforms xxxviii
Using the Indexes xxxix
Using the Examples in This Manual xxxix
Merging a Full Example xxxix
Merging a Snippet xl
Documentation Conventions xl
Documentation Feedback xlii
Requesting Technical Support xlii
Self-Help Online Tools and Resources xliii
Opening a Case with JTAC xliii

Part 1 Overview

Chapter 1 Routing Protocols Concepts 3

Routing Databases Overview 3
Routing Protocol Databases 3
JUNOS Routing Tables 4
Forwarding Tables 5
How the Routing and Forwarding Tables Are Synchronized 5
Route Preferences Overview 6
Alternate and Tiebreaker Preferences 6
Multiple Active Routes 6
How the Active Route Is Determined 7
Default Route Preference Values 8
Equal-Cost Paths and Load Sharing 10
IPv6 Overview 10
IPv6 Packet Headers 11
Header Structure 11
Extension Headers 11
IPv6 Addressing 12
Address Representation 12
Address Types 12

Address Scope 13
Address Structure 13
IPv6 Standards 13

Chapter 2 Complete Routing and Routing Protocol Configuration Statements 15

[edit logical-systems] Hierarchy Level 15
[edit protocols] Hierarchy Level 16
[edit routing-instances] Hierarchy Level 33
[edit routing-options] Hierarchy Level 38

Part 2 Protocol-Independent Routing Properties

Chapter 3 Protocol-Independent Routing Properties Overview 45

Protocol-Independent Routing Properties Configuration Statements 45
Minimum Protocol-Independent Routing Properties Configuration 49

Chapter 4 Configuring Routing Tables and Routes 51

Creating Routing Tables 52
Example: Creating Routing Tables 53
Configuring Static Routes 54
Configuring the Destination of Static Routes 57
Configuring the Next Hop for Static Routes 57
Configuring an Independent Preference for Static Routes 58
Example: Configuring Independent Preferences for an IPv4 Static Route 60
Example: Configuring Independent Preferences for an IPv6 Static Route 60
Example: Configuring Independent Preferences for an Unnumbered Ethernet Interface 61
Specifying an LSP as the Next Hop for Static Routes 62
Installing Static Routes into More than One Routing Table 63
Examples: Installing a Static Route into More than One Routing Table 63
Configuring CLNS Static Routes 64
Example: Configuring a Static CLNS Route 64
Configuring Static Route Options 65
Configuring a Metric Value for Static Routes 67
Configuring a Preference Value for Static Routes 68
Associating BGP Communities with Static Routes 68
Associating AS Paths with Static Routes 69
Configuring an OSPF Tag String for Static Routes 70
Controlling Temporary Installation of Static Routes in the Forwarding Table 71

Controlling Retention of Static Routes in the Forwarding Table 71
Controlling Retention of Inactive Static Routes in the Routing and Forwarding Tables 72
Controlling Readvertisement of Static Routes 73
Controlling Resolution of Static Routes to Prefixes That Are Not Directly Connected 74
Configuring Bidirectional Forwarding Detection 74
Tracing BFD Protocol Traffic 78
Overview of BFD Authentication for Static Routes 79
BFD Authentication Algorithms 80
Security Authentication Keychains 81
Strict Versus Loose Authentication 81
Configuring BFD Authentication for Static Routes 81
Configuring the BFD Authentication Parameters 82
Viewing Authentication Information for BFD Sessions 83
Configuring Default Routes 84
Propagating Static Routes into Routing Protocols 85
Examples: Configuring Static Routes 85
Configuring Aggregate Routes 87
Configuring the Destination of Aggregate Routes 89
Configuring Aggregate Route Options 89
Configuring a Metric Value for Aggregate Routes 90
Configuring a Preference Value for Aggregate Routes 90
Configuring the Next Hop for Aggregate Routes 91
Associating BGP Communities with Aggregate Routes 91
Associating AS Paths with Aggregate Routes 92
Including AS Numbers in Aggregate Route Paths 93
Configuring an OSPF Tag String for Aggregate Routes 93
Controlling Retention of Inactive Aggregate Routes in the Routing and Forwarding Tables 94
Applying Policies to Aggregate Routes 94
Advertising Aggregate Routes 95
Configuring Generated Routes 96
Configuring the Destination of Generated Routes 97
Configuring Generated Route Options 97
Configuring a Metric Value for Generated Routes 98
Configuring a Preference Value for Generated Routes 99
Configuring the Next Hop for Generated Routes 99
Associating BGP Communities with Generated Routes 99
Associating AS Paths with Generated Routes 100
Configuring an OSPF Tag String for Generated Routes 101
Including AS Numbers in Generated Route Paths 101
Controlling Retention of Inactive Generated Routes in the Routing and Forwarding Tables 102
Applying Policies to Generated Routes 102
Configuring Martian Addresses 103
Adding Martian Addresses 103
Deleting Martian Addresses 104
Using Class E Addresses for Interface Addresses 105

Configuring Flow Routes 105
Configuring Match Conditions for Flow Routes 106
Configuring the Action for Flow Routes 108
Validating Flow Routes 108
Enabling Support for BGP Flow-Specification Algorithm Version 7 and Later 109
Applying Filters to the Forwarding Table 109

Chapter 5 Configuring Other Protocol-Independent Routing Properties 111

Configuring AS Numbers for BGP 112
Configuring Router Identifiers for BGP and OSPF 113
Configuring AS Confederation Members 113
Configuring Route Recording for Flow Aggregation 114
Creating Routing Table Groups 114
Examples: Creating Routing Table Groups 116
Configuring How Interface Routes Are Imported into Routing Tables 116
Configuring Multicast Scoping 117
Example: Configuring Multicast Scoping 118
Enabling Multicast Forwarding Without PIM 118
Configuring Additional Source-Specific Multicast Groups 119
Configuring Multicast Forwarding Cache Limits 119
Configuring Per-Packet Load Balancing 120
Examples: Configuring Per-Packet Load Balancing 121
Configuring Unicast Reverse-Path-Forwarding Check 122
Example: Configuring Unicast RPF 123
Configuring Graceful Restart 124
Configuring Route Distinguishers for VRF and Layer 2 VPN Instances 125
Configuring Dynamic GRE Tunnels for VPNs 125
Configuring System Logging for the Routing Protocol Process 126
Examples: Configuring System Logging for the Routing Protocol Process 127
Configuring Route Resolution 127
Enabling Indirect Next Hops 128
Enabling Nonstop Active Routing 129
Tracing Global Routing Protocol Operations 129
Examples: Tracing Global Routing Protocol Operations 131
Disabling Distributed Periodic Packet Management on the Packet Forwarding Engine 131
Enabling Source Routing 132
Delaying Updates of the MED Path Attribute for BGP 133
Creating Policies to Control Label Allocation and Substitution for MPLS Ingress and AS Border Routers 133

Chapter 6 Configuring Logical Systems 135

Logical Systems Overview 135
Logical System Configuration Statements 137
Minimum Logical System Configuration 138
Configuring a Logical System 138
logical-systems 139

Chapter 7 Summary of Protocol-Independent Routing Properties Configuration Statements 141

active 142
aggregate 143
as-path 146
auto-export 148
autonomous-system 149
bfd 151
bfd-liveness-detection 154
brief 158
color 158
community 159
confederation 161
destination-networks 161
disable 162
discard 163
dynamic-tunnels 164
export 164
export-rib 165
fate-sharing 166
filter 167
flow 168
forwarding-cache 169
forwarding-table 169
full 170
generate 171
graceful-restart 172
import 173
import-policy 173
import-rib 174
independent-domain 175
indirect-next-hop 176
input 176
install 177
instance-export 178
instance-import 178
interface 179
    interface (Multicast via Static Routes) 180
    interface (Multicast Scoping) 181
interface-routes 182

lsp-next-hop 183
martians 184
maximum-paths 185
maximum-prefixes 186
med-igp-update-interval 187
metric 188
    metric (Aggregate, Generated, or Static Route) 188
    metric (Qualified Next Hop on Static Route) 189
multicast 190
no-install 190
no-readvertise 191
no-retain 191
nonstop-routing 191
options 192
p2mp-lsp-next-hop 193
passive 193
policy 194
ppm 195
preference 196
prefix 197
qualified-next-hop 198
readvertise 199
resolution 200
resolution-ribs 200
resolve 201
restart-duration 202
retain 203
rib 204
    rib (General) 205
    rib (Route Resolution) 207
rib-group 208
rib-groups 209
route-distinguisher-id 210
route-record 210
router-id 211
routing-options 211
scope 212
source-address 213
source-routing 213
ssm-groups 214
static 215
tag 218
threshold 219
traceoptions 220
tunnel-type 222
unicast-reverse-path 222

Part 3 Routing Instances

Chapter 8 Introduction to Routing Instances 225

Routing Instances Overview 225

Chapter 9 Routing Instances Configuration Guidelines 229

Complete Routing Instances Configuration Statements 229
Routing Instances Minimum Configuration 234
Minimum Routing-Instance Configuration for BGP 234
Minimum Routing-Instance Configuration for IS-IS 235
Minimum Routing-Instance Configuration for Layer 2 VPNs 235
Minimum Routing-Instance Configuration for LDP 236
Minimum Routing-Instance Configuration for MSDP 236
Minimum Routing-Instance Configuration for Multiprotocol BGP-Based Multicast VPNs 237
Minimum Routing-Instance Configuration for OSPF 237
Minimum Routing-Instance Configuration for OSPFv3 238
Minimum Routing-Instance Configuration for PIM 238
Minimum Routing-Instance Configuration for RIP 239
Minimum Routing-Instance Configuration for VPLS 239
Configuring Multiple Instances of BGP 240
Example: Configuring Multiple Instances of BGP 240
Configuring Multiple Instances of IS-IS 241
Example: Configuring Multiple Routing Instances of IS-IS 242
Configuring Multiple Instances of LDP 245
Configuring Multiple Instances of MSDP 246
Configuring Multiple Instances of OSPF 247
Example: Configuring Multiple Routing Instances of OSPF 247
Configuring Multiple Instances of PIM 250
Configuring Multiple Instances of RIP 251
Configuring Routing Instances 251
Specifying the Instance Type for Routing Instances 253
Configuring VRF Routing Instances 254
Configuring Non-VPN VRF Routing Instances 255
Configuring VPLS Routing Instances 256
Configuring Route Distinguishers for Routing Instances 257
Configuring Filter-Based Forwarding 258
Configuring Class-of-Service-Based Forwarding 259
Configuring Secondary VRF Import and Export Policy 260
Configuring Policy-Based Export for Routing Instances 261
Example: Configuring Policy-Based Export for an Overlapping VPN 261
Example: Configuring Policy-Based Export for a Nonforwarding Instance 263
Configuring VRF Table Labels 265
Configuring VRF Targets 265
Configuring OSPF Domain IDs for VPNs 266
Examples: Configuring an OSPF Domain ID 269

Configuring Route Limits for Routing Tables 271
Configuring Independent AS Domains 271

Chapter 10 Summary of Routing Instances Configuration Statements 273

access-profile 273
description 274
forwarding-options 274
instance-type 275
interface 277
no-vrf-advertise 277
protocols 278
route-distinguisher 281
routing-instances 282
routing-options 282
vrf-export 283
vrf-import 283
vrf-table-label 284
vrf-target 284

Part 4 Multitopology Routing

Chapter 11 Introduction to Multitopology Routing 287

Multitopology Routing Overview 287
Routing Table Naming Conventions for Multitopology Routing 287
Routing Protocol Support for Multitopology Routing 288
Filter-Based Forwarding Support 288
Multitopology Routing Standards 289

Chapter 12 Multitopology Routing Configuration Guidelines 291

Configuring Topologies 291
Configuring Multitopology Routing in OSPF 292
Configuring Topologies and SPF Options for MT-OSPF 292
Configuring a Prefix Export Limit for MT-OSPF 294
Configuring a Topology to Appear Overloaded 294
Configuring Interface Properties for MT-OSPF 294
Disabling MT-OSPF on OSPF Interfaces 295
Disabling MT-OSPF on Virtual Links 295
Advertising MPLS Label-Switched Paths into MT-OSPF 296
Configuring Other MT-OSPF Properties 297
Configuring Multitopology Routing in Static Routes 298
Configuring Multitopology Routing in BGP 299
BGP Route Resolution in Multitopology Routing 299
Configuring Filter-Based Forwarding for Multitopology Routing 300

Chapter 13 Summary of Multitopology Routing Configuration Statements 303

community 304
rib 305
topologies 306
topology 307
topology (Filter-Based Forwarding) 308
topology (Multitopology Routing) 309
topology (OSPF) 310
topology (OSPF Interface) 311
topology-id 312

Part 5 Interior Gateway Protocols

Chapter 14 Introduction to IS-IS 315

IS-IS Overview 315
IS-IS Terminology 315
ISO Network Addresses 316
IS-IS Packets 317
Persistent Route Reachability 317
IS-IS Extensions to Support Traffic Engineering 317
IS-IS IGP Shortcuts 318
IS-IS Extensions to Support Route Tagging 318
IS-IS Standards 318

Chapter 15 IS-IS Configuration Guidelines 321

Configuring IS-IS 322
Minimum IS-IS Configuration 324
Configuring IS-IS Authentication 325
Configuring of Interface-Specific IS-IS Properties 327
Configuring BFD for IS-IS 328
Overview of BFD Authentication for IS-IS 331
BFD Authentication Algorithms 331
Security Authentication Keychains 332
Strict Versus Loose Authentication 332
Configuring BFD Authentication for IS-IS 333
Configuring BFD Authentication Parameters 333
Viewing Authentication Information for BFD Sessions 334
Enabling Packet Checksum on IS-IS Interfaces 336
Configuring the Transmission Frequency for CSNP Packets on IS-IS Interfaces 336
Configuring Synchronization Between LDP and IS-IS 336
Configuring the Transmission Frequency for Link-State PDUs on IS-IS Interfaces 337

Configuring Mesh Groups of IS-IS Interfaces 337
Configuring IS-IS Multicast Topologies 338
Example: Configuring IS-IS Multicast Topologies 339
Configuring IS-IS IPv6 Unicast Topologies 340
Configuring Point-to-Point Interfaces for IS-IS 340
Configuring Levels on IS-IS Interfaces 341
Disabling IS-IS at a Level on IS-IS Interfaces 342
Example: Disabling IS-IS at a Level 342
Advertising Interface Addresses Without Running IS-IS 343
Configuring Authentication for IS-IS Hello Packets 343
Configuring the Transmission Frequency for IS-IS Hello Packets 343
Configuring the Delay Before IS-IS Neighbors Mark the Routing Device as Down 344
Configuring the Metric Value for IS-IS Routes 344
Configuring the IS-IS Metric Value Used for Traffic Engineering 344
Configuring the Designated Router Priority for IS-IS 345
Advertising Interface Addresses Without Running IS-IS 345
Configuring the Reference Bandwidth Used in IS-IS Metric Calculations 345
Limiting the Number of Advertised IS-IS Areas 346
Enabling Wide IS-IS Metrics for Traffic Engineering 346
Configuring Preference Values for IS-IS Routes 347
Limiting the Number of Prefixes Exported to IS-IS 347
Configuring Link-State PDU Lifetime for IS-IS 347
Advertising Label-Switched Paths into IS-IS 348
Configuring IS-IS to Make Routing Devices Appear Overloaded 348
Configuring SPF Options for IS-IS 349
Configuring Graceful Restart for IS-IS 350
Configuring IS-IS for Multipoint Network Clouds 351
Configuring IS-IS Traffic Engineering Attributes 351
Configuring IS-IS to Use IGP Shortcuts 351
Configuring IS-IS to Ignore the Metric of RSVP Label-Switched Paths 352
Disabling IS-IS Support for Traffic Engineering 353
Installing IPv4 Routes into the Multicast Routing Table 353
Configuring IS-IS to Use Protocol Preference to Determine the Traffic Engineering Database Credibility Value 354
Enabling Authentication for IS-IS Without Network-Wide Deployment 354
Configuring Quicker Advertisement of IS-IS Adjacency State Changes 355
Enabling Padding of IS-IS Hello Packets 355
Configuring CLNS for IS-IS 356
Example: Configuring CLNS for IS-IS 357
Disabling IS-IS 358
Disabling IPv4 Routing for IS-IS 359
Disabling IPv6 Routing for IS-IS 359
Applying Policies to Routes Exported to IS-IS 360
Examples: Configuring IS-IS Routing Policy 360
Installing a Default Route to the Nearest Routing Device That Operates at Both IS-IS Levels 362
Configuring Loop-Free Alternate Routes for IS-IS 363
Configuring Link Protection for IS-IS 365
Configuring Node-Link Protection for IS-IS 365
Excluding an IS-IS Interface as a Backup for Protected Interfaces 366

Configuring RSVP Label-Switched Paths as Backup Paths for IS-IS 366
Using Operational Mode Commands to Monitor Protected IS-IS Routes 367
Example: Configuring Node-Link Protection for IS-IS Routes 367
Tracing IS-IS Protocol Traffic 369
Examples: Tracing IS-IS Protocol Traffic 371

Chapter 16 Summary of IS-IS Configuration Statements 373

authentication-key 374
authentication-type 375
bfd-liveness-detection 376
checksum 378
clns-routing 378
csnp-interval 379
disable 380
disable (IS-IS) 381
disable (LDP Synchronization) 382
export 382
external-preference 383
family 384
graceful-restart 385
hello-authentication-key 386
hello-authentication-type 387
hello-interval 388
hello-padding 389
hold-time 390
hold-time (IS-IS) 390
hold-time (LDP Synchronization) 391
ignore-attached-bit 391
ignore-lsp-metrics 392
interface 393
ipv4-multicast 395
ipv4-multicast-metric 395
ipv6-multicast 396
ipv6-multicast-metric 396
ipv6-unicast 397
ipv6-unicast-metric 397
isis 398
label-switched-path 399
ldp-synchronization 400
level 401
level (Global IS-IS) 402
level (IS-IS Interfaces) 403
link-protection 404
loose-authentication-check 405
lsp-interval 405
lsp-lifetime 406
max-areas 406
mesh-group 407

metric 408
multicast-rpf-routes 409
no-adjacency-holddown 409
no-authentication-check 410
no-csnp-authentication 410
no-eligible-backup 411
no-hello-authentication 411
no-ipv4-multicast 412
no-ipv4-routing 412
no-ipv6-multicast 413
no-ipv6-routing 413
no-ipv6-unicast 414
no-psnp-authentication 414
no-unicast-topology 415
node-link-protection 415
overload 416
passive 417
point-to-point 418
preference 418
prefix-export-limit 419
priority 420
reference-bandwidth 421
rib-group 422
shortcuts 423
spf-options 424
te-metric 425
topologies 426
traceoptions 427
traffic-engineering 429
wide-metrics-only 430

Chapter 17 Introduction to OSPF 431

OSPF Overview 432
OSPF Routing Algorithm 433
OSPF Version 3 434
Understanding OSPF Areas 434
Areas 435
Area Border Routers 435
Backbone Areas 435
AS Boundary Routers 435
Stub Areas 435
Not-So-Stubby Areas 436
Transit Areas 436
Overview of Packets 436
OSPF Packet Header 436
Hello Packets 437
Database Description Packets 437
Link-State Request Packets 438
Link-State Update Packets 438

Link-State Acknowledgment Packets 438
Link-State Advertisement Packet Types 438
OSPF External Metrics Overview 439
OSPF Designated Router Overview 439
OSPF Extensions to Support Traffic Engineering 439
Configuring OSPF IGP Shortcuts 440
OSPF Standards 440

Chapter 18 OSPF Configuration Guidelines 443

Configuring OSPF 444
Minimum OSPF Configuration 449
Configuring OSPF Areas 449
Configuring the OSPF Backbone Area 450
Configuring OSPF Nonbackbone Areas 450
Configuring OSPF Stub Areas 450
Configuring OSPF Not-So-Stubby Areas 451
Configuring OSPF Virtual Links 452
Example: Configuring an OSPF Virtual Link 452
Disabling Export of LSAs into NSSAs Attached to ASBR ABRs 453
Disabling OSPFv2 Compatibility with RFC 1583 453
Configuring OSPF on Interfaces 454
Configuring an Interface on a Broadcast or Point-to-Point Network 454
Configuring an Interface on a Point-to-Multipoint Network 455
Configuring an Interface on a Nonbroadcast, Multiaccess Network 455
Configuring an OSPF Demand Circuit Interface 456
Configuring Multiarea Adjacency in OSPFv2 457
Configuring Multiple Address Families for OSPFv3 457
Configuring Authentication for OSPFv2 458
Example: Configuring IPsec Authentication for an OSPFv2 Interface 460
Example: Configuring a Transition of MD5 Keys 461
Example: Configuring MD5 Authentication 461
Configuring Authentication for OSPFv3 462
Limiting the Number of Prefixes Exported to OSPF 462
Configuring the Designated Router Priority for OSPF 462
Summarizing Ranges of Routes in OSPF Link-State Advertisements 463
Configuring the Metric Value for OSPF Interfaces 464
Dynamically Adjusting OSPF Interface Metrics Based on Bandwidth 465
Configuring Preference Values for OSPF Routes 465
Configuring OSPF Timers 466
Modifying the Hello Interval 466
Controlling the LSA Retransmission Interval 467
Modifying the Router Dead Interval 467
Specifying the Transit Delay 467
Configuring OSPF Refresh and Flooding Reduction in Stable Topologies 468

Configuring BFD for OSPF 469
Overview of BFD Authentication for OSPF 471
BFD Authentication Algorithms 472
Security Authentication Keychains 472
Strict Versus Loose Authentication 473
Configuring BFD Authentication for OSPF 473
Configuring BFD Authentication Parameters 473
Viewing Authentication Information for BFD Sessions 475
Configuring Synchronization Between LDP and IGPs 476
Configuring Graceful Restart for OSPF and OSPFv3 477
Configuring SPF Options for OSPF 478
Advertising Interface Addresses Without Running OSPF 479
Configuring OSPF Passive Traffic Engineering Mode 479
Advertising Label-Switched Paths into OSPF 480
Configuring OSPF to Make Routing Devices Appear Overloaded 481
Enabling OSPF Traffic Engineering Support 481
Example: Enabling OSPF Traffic Engineering Support 483
Configuring the OSPF Metric Value Used for Traffic Engineering 484
Applying Policies to OSPF Routes 484
Configuring Import and Export Policies for Network Summaries 485
Configuring Priority for Prefixes in Import Policy 486
Example: Configuring a Route Filter Policy to Specify Priority for Prefixes Learned Through OSPF 486
Configuring OSPF Routing Table Groups 487
Configuring OSPF Sham Links 488
Configuring OSPF Peer Interfaces 488
Loop-Free Alternate Routes for OSPF Overview 489
Configuring Link Protection for OSPF 490
Configuring Node-Link Protection for OSPF 491
Excluding an OSPF Interface as a Backup for a Protected Interface 492
Configuring Backup SPF Options for Protected OSPF Interfaces 493
Configuring RSVP Label-Switched Paths as Backup Paths for OSPF 495
Tracing OSPF Protocol Traffic 495
Examples: Tracing OSPF Protocol Traffic 497

Chapter 19 Summary of OSPF Configuration Statements 499

area 500
area-range 501
authentication 503
backup-spf-options 504
bandwidth-based-metrics 505
bfd-liveness-detection 507
dead-interval 511
default-lsa 512
default-metric 513
demand-circuit 514
disable 515
disable (LDP Synchronization) 516
disable (OSPF) 517

domain-id 518
domain-vpn-tag 518
export 519
external-preference 520
flood-reduction 521
graceful-restart 522
hello-interval 523
hold-time 524
ignore-lsp-metrics 524
import 525
inter-area-prefix-export 526
inter-area-prefix-import 527
interface 528
interface-type 530
ipsec-sa 531
label-switched-path 532
ldp-synchronization 533
link-protection 534
lsp-metric-into-summary 535
md5 536
metric 537
metric-type 539
neighbor 540
network-summary-export 541
network-summary-import 541
no-eligible-backup 542
no-nssa-abr 543
no-rfc-1583 544
no-summaries 544
node-link-protection 545
nssa 546
ospf 547
ospf3 547
overload 548
passive 550
peer-interface 552
poll-interval 553
preference 554
prefix-export-limit 555
priority 556
realm 557
reference-bandwidth 558
retransmit-interval 559
rib-group 560
route-type-community 561
secondary 561
sham-link 562
sham-link-remote 562
shortcuts 563
simple-password 564
spf-options 565

stub 567
summaries 568
te-metric 569
traceoptions 570
traffic-engineering 573
traffic-engineering (OSPF) 574
traffic-engineering (Passive TE Mode) 575
transit-delay 576
transmit-interval 577
type-7 578
virtual-link 579

Chapter 20 Introduction to RIP 581

RIP Overview 581
RIP Protocol Overview 581
RIP Packets 582
RIP Standards 582

Chapter 21 RIP Configuration Guidelines 583

Configuring RIP 583
Minimum RIP Configuration 585
Overview of RIP Global Properties 586
Overview of RIP Neighbor Properties 586
Configuring Authentication for RIP 587
Configuring BFD for RIP 588
Overview of BFD Authentication for RIP 590
BFD Authentication Algorithms 590
Security Authentication Keychains 591
Strict Versus Loose Authentication 591
Configuring BFD Authentication for RIP 592
Configuring BFD Authentication Parameters 592
Viewing Authentication Information for BFD Sessions 594
Accepting RIP Packets with Nonzero Values in Reserved Fields 595
Applying Policies to RIP Routes Imported from Neighbors 596
Configuring the Number of Route Entries in RIP Update Messages 596
Configuring the Metric Value Added to Imported RIP Routes 596
Configuring RIP Update Messages 597
Configuring Routing Table Groups for RIP 597
Configuring RIP Timers 597
Configuring Group-Specific RIP Properties 598
Applying Policies to Routes Exported by RIP 599
Configuring the Default Preference Value for RIP 599
Configuring the Metric for Routes Exported by RIP 600
Configuring Graceful Restart for RIP 600
Disabling Strict Address Checking for RIP Messages 600
Tracing RIP Protocol Traffic 601
Example: Tracing RIP Protocol Traffic 602
Example: Configuring RIP 603

Chapter 22 Summary of RIP Configuration Statements 605

any-sender 605
authentication-key 606
authentication-type 607
bfd-liveness-detection 608
check-zero 610
export 611
graceful-restart 612
group 613
holddown 615
import 616
message-size 617
metric-in 618
metric-out 619
neighbor 620
no-check-zero 621
preference 622
receive 623
rib-group 624
rip 624
route-timeout 625
send 626
traceoptions 627
update-interval 629

Chapter 23 Introduction to RIPng 631

RIPng Overview 631
RIPng Protocol Overview 631
RIPng Packets 632
RIPng Standards 632

Chapter 24 RIPng Configuration Guidelines 633

Configuring RIPng 633
Minimum RIPng Configuration 634
Overview of RIPng Global Properties 635
Overview of RIPng Neighbor Properties 635
Applying Policies to RIPng Routes Imported from Neighbors 636
Configuring the Metric Value Added to Imported RIPng Routes 636
Configuring RIPng Update Messages 636
Configuring RIPng Timers 637
Configuring Group-Specific RIPng Properties 637
Applying Policies to Routes Exported by RIPng 638
Configuring the Default Preference Value for RIPng 638
Configuring the Metric for Routes Exported by RIPng 639

Configuring Graceful Restart for RIPng 639
Tracing RIPng Protocol Traffic 639
Example: Configuring RIPng 641

Chapter 25 Summary of RIPng Configuration Statements 643

export 643
graceful-restart 644
group 645
holddown 646
import 647
metric-in 648
metric-out 649
neighbor 650
preference 651
receive 652
ripng 653
route-timeout 653
send 654
traceoptions 655
update-interval 657

Chapter 26 Introduction to ICMP Router Discovery 659

ICMP Router Discovery Overview 659
Operation of a Router Discovery Server 659
Router Advertisement Messages 660
ICMP Router Discovery Standards 660

Chapter 27 ICMP Router Discovery Configuration Guidelines 661

Configuring ICMP Router Discovery 661
Minimum ICMP Router Discovery Configuration 662
Configuring the Addresses Included in ICMP Router Advertisements 662
Configuring the Frequency of ICMP Router Advertisements 663
Modifying the Lifetime in ICMP Router Advertisements 663
Tracing ICMP Protocol Traffic 663
Example: Tracing ICMP Protocol Traffic 664

Chapter 28 Summary of ICMP Router Discovery Configuration Statements 665

address 665
advertise 666
broadcast 666
disable 667
ignore 667
ineligible 667
interface 668

lifetime 669
max-advertisement-interval 670
min-advertisement-interval 671
multicast 672
priority 673
router-discovery 673

Chapter 29 Introduction to Neighbor Discovery 675

Neighbor Discovery Overview 675
Router Discovery 676
Address Resolution 676
Redirect 676
Neighbor Discovery Standards 676

Chapter 30 Neighbor Discovery Configuration Guidelines 677

Configuring Neighbor Discovery 677
Minimum Neighbor Discovery Configuration 678
Configuring an Interface to Send Neighbor Discovery Advertisements 678
Configuring the Hop Count in Outgoing Neighbor Discovery Packets 679
Configuring the Lifetime for the Default Neighbor Discovery Router 679
Enabling Stateful Autoconfiguration with Neighbor Discovery 679
Configuring the Frequency of Neighbor Discovery Advertisements 680
Configuring the Delay Before Neighbor-Discovery Neighbors Mark the Router as Down 680
Configuring the Frequency of Neighbor Solicitation Messages 681
Configuring the Prefix Information Included in Neighbor Discovery Advertisements 681
Setting the Prefix for Onlink Determination 681
Setting the Prefix for Stateless Address Autoconfiguration 682
Configuring the Preferred Lifetime 682
Configuring the Valid Lifetime 682
Tracing Neighbor Discovery Protocol Traffic 683

Chapter 31 Summary of Neighbor Discovery Router Advertisement Configuration Statements 685

autonomous 685
current-hop-limit 686
default-lifetime 686
interface 687
managed-configuration 688
max-advertisement-interval 688
min-advertisement-interval 689
no-autonomous 689
no-managed-configuration 689
no-on-link 689
no-other-stateful-configuration 689

on-link 690
other-stateful-configuration 690
preferred-lifetime 691
prefix 692
reachable-time 692
retransmit-timer 693
router-advertisement 693
traceoptions 694
traceoptions 697
valid-lifetime 698

Chapter 32 Secure Neighbor Discovery Configuration Guidelines 699

Secure Neighbor Discovery Configuration Overview 699
Configuring Secure Neighbor Discovery 699
Enabling Secure Neighbor Discovery 700
Configuring Cryptographically Generated Addresses for Secure Neighbor Discovery 700
Specifying the Pathname for the Key File 701
Specifying the RSA Key Length 701
Configuring Timestamps for Secure Neighbor Discovery 701
Tracing Secure Neighbor Discovery Protocol Traffic 702

Chapter 33 Summary of Secure Neighbor Discovery Configuration Statements 703

cryptographic-address 703
key-length 704
key-pair 704
neighbor-discovery 705
secure 706
security-level 707
timestamp 708
traceoptions 709

Part 6 BGP

Chapter 34 Introduction to BGP 713

BGP Overview 714
Autonomous Systems 714
AS Paths and Attributes 714
External and Internal BGP 715
BGP Routes Overview 715
Overview of BGP Messages 716
Open Messages 717
Update Messages 717

Keepalive Messages 718
Notification Messages 718
BGP Standards 718

Chapter 35 BGP Configuration Guidelines 721

Configuring BGP 722
Minimum BGP Configuration 723
Enabling BGP 724
Specifying the Local Routing Device's AS Number 724
Defining an AS Confederation and Its Members 724
Assigning a BGP Identifier 725
Defining BGP Global Properties 725
Configuring BGP Groups and Peers 728
Defining a Group with Static Peers 728
Example: Defining a Large Number of Groups with Static Peers 729
Example: Defining a Small Number of Groups with Static Peers for Better Scalability 730
Defining a Group with Dynamic Peers 730
Defining the Group Type 731
Specifying the Peer's AS Number 731
Defining Group Properties 732
Defining Peer Properties 734
Examples: Configuring BGP Groups, Peers, and Confederations 737
Configuring the Delay Before BGP Peers Mark the Routing Device as Down 741
Configuring MTU Discovery for BGP Sessions 741
Configuring Graceful Restart for BGP 742
Advertising Explicit Null Labels to BGP Peers 742
Configuring Aggregate Labels for VPNs 743
Configuring Authentication for BGP 743
Using IPsec to Protect BGP Traffic 745
Disabling Transmission of Open Requests to BGP Peers 746
Configuring a Local Endpoint Address for BGP Sessions 746
Configuring the MED in BGP Updates 746
Defining a MED Metric Directly 747
Using Routing Policy to Define a MED Metric 748
Examples: Configuring the MED Metric 748
Controlling BGP Route Aggregation 750
Configuring EBGP Multihop Sessions 750
Configuring Single-Hop EBGP Peers to Accept Remote Next Hops 750
Example: Configure an Import Routing Policy for an EBGP Peer to Accept a Remote Next Hop 751
Configuring the Local Preference Value for BGP Routes 752
Configuring the Default Preference Value for BGP Routes 752
Examples: Configuring BGP Route Preference 753
Configuring Routing Table Path Selection for BGP 754
Example: Always Comparing MEDs 755
Selecting Multiple Equal-Cost Active Paths 755

Configuring a Local AS for EBGP Sessions 756
Examples: Configuring a Local AS 758
Removing Private AS Numbers from AS Paths 761
Configuring BGP Route Reflection 761
Examples: Configuring BGP Route Reflection 763
Configuring Flap Damping for BGP Routes 767
Enabling Multiprotocol BGP 767
Limiting the Number of Prefixes Received on a BGP Peering Session 771
Limiting the Number of Prefixes Accepted on a BGP Peering Session 771
Configuring BGP Routing Table Groups 772
Resolving Routes to PE Routing Devices Located in Other ASs 772
Allowing Labeled and Unlabeled Routes 773
Enabling BGP to Carry Flow-Specification Routes 773
Configuring Flow-Specification Routes for IPv4 Unicast 773
Configuring Flow-Specification Routes for Layer 3 VPNs 774
Enabling BGP to Carry CLNS Routes 774
Example: Enabling CLNS Between Two Routers 775
Example: Configuring CLNS Within a VPN 777
Enabling BGP Route Target Filtering 779
Applying Filters Provided by BGP Peers to Outbound Routes 780
Enabling Layer 2 VPN and VPLS Signaling 781
Applying Policies to BGP Routes 781
Applying Routing Policy 782
Applying Policies to Routes Being Imported into the Routing Table from BGP 782
Applying Policies to Routes Being Exported from the Routing Table into BGP 783
Setting BGP to Advertise Inactive Routes 783
Configuring BGP to Advertise the Best External Route to Internal Peers 783
Configuring How Often BGP Exchanges Routes with the Routing Table 784
Disabling Suppression of Route Advertisements 785
Preventing Automatic Reestablishment of BGP Peering Sessions After NSR Switchovers 786
Configuring EBGP Peering Using IPv6 Link-Local Addresses 786
Configuring IPv6 BGP Routes over IPv4 Transport 787
Example: Configuring IPv6 BGP Routes over IPv4 Transport 787
Configuring System Logging of BGP Peer State Transitions 788
Configuring a Text Description for BGP Groups or Peers 788
Restricting TCP Connections to BGP Peers 789
Applying BGP Export Policy to VRF Routes 789
Including Next-Hop Reachability Information in Multiprotocol Updates 789
Configuring BFD for BGP 789
Overview of BFD Authentication for BGP 792
BFD Authentication Algorithms 793
Security Authentication Keychains 793
Strict Versus Loose Authentication 794
Configuring BFD Authentication for BGP 794
Configuring BFD Authentication Parameters 795
Viewing Authentication Information for BFD Sessions 796

Limiting TCP Segment Size for BGP 798
Configuring the BGP Monitoring Protocol 798
Tracing BGP Protocol Traffic 799
Examples: Tracing BGP Protocol Traffic 800

Chapter 36 Summary of BGP Configuration Statements 803

accept-remote-nexthop 804
accepted-prefix-limit 805
advertise-external 807
advertise-inactive 808
advertise-peer-as 809
aggregate-label 810
allow 811
as-override 812
authentication-algorithm 813
authentication-key 814
authentication-key-chain 815
bfd-liveness-detection 816
bgp 820
bgp-orf-cisco-mode 821
bmp 822
cluster 823
damping 824
description 825
disable 826
explicit-null 827
export 828
family 829
flow 832
graceful-restart 833
group 834
hold-time 837
idle-after-switch-over 838
import 839
include-mp-next-hop 840
ipsec-sa 840
iso-vpn 841
keep 842
labeled-unicast 843
local-address 844
local-as 845
local-interface 847
local-preference 848
log-updown 849
metric-out 850
mtu-discovery 852
multihop 853
multipath 854
neighbor 855

no-advertise-peer-as 857
no-aggregator-id 858
no-client-reflect 859
no-validate 860
out-delay 861
outbound-route-filter 862
passive 863
path-selection 864
peer-as 865
preference 866
prefix-limit 867
remove-private 869
resolve-vpn 870
rib 871
rib-group 872
route-target 873
tcp-mss 874
traceoptions 875
type 878
vpn-apply-export 878

Part 7 Indexes

Index 881
Index of Statements and Commands 901

List of Figures

Part 1 **Overview**

Chapter 1 **Routing Protocols Concepts** **3**

Figure 1: Synchronizing Routing Exchange Between the Routing and Forwarding Tables 5

Part 2 **Protocol-Independent Routing Properties**

Chapter 5 **Configuring Other Protocol-Independent Routing Properties** **111**

Figure 2: Route to Forwarding Next-Hop Bindings 128

Figure 3: Route to Forwarding Indirect Next-Hop Bindings 128

Part 3 **Routing Instances**

Chapter 9 **Routing Instances Configuration Guidelines** **229**

Figure 4: Configuration for Multiple Routing Instances 242

Figure 5: Configuration for Multiple Routing Instances 248

Figure 6: Configuration of Policy-Based Export for an Overlapping VPN 262

Part 5 **Interior Gateway Protocols**

Chapter 15 **IS-IS Configuration Guidelines** **321**

Figure 7: Install Default Route to Nearest Routing Device That Operates at Both Level 1 and Level 2 362

Figure 8: Link Protection and Node-Link Protection Comparison for IS-IS Routes 364

Part 6 **BGP**

Chapter 34 **Introduction to BGP** **713**

Figure 9: ASs, EBGP, and IBGP 715

Chapter 35 **BGP Configuration Guidelines** **721**

Figure 10: Example: BGP Confederation Topology 738

Figure 11: Local AS Configuration 759

Figure 12: Simple Route Reflector 764

List of Tables

About This Guide **xxxvii**
Table 1: Notice Icons xli
Table 2: Text and Syntax Conventions xli

Part 1 Overview

Chapter 1 Routing Protocols Concepts **3**
Table 3: Default Route Preference Values 8

Part 2 Protocol-Independent Routing Properties

Chapter 4 Configuring Routing Tables and Routes **51**
Table 4: Flow Route Match Conditions 106
Table 5: Flow Route Action Modifiers 108

Part 4 Multitopology Routing

Chapter 11 Introduction to Multitopology Routing **287**
Table 6: Examples of Routing Tables for Custom Topologies 287

About This Guide

This preface provides the following guidelines for using the *JUNOS® Software Routing Protocols Configuration Guide*:

- JUNOS Documentation and Release Notes on page xxxvii
- Objectives on page xxxviii
- Audience on page xxxviii
- Supported Platforms on page xxxviii
- Using the Indexes on page xxxix
- Using the Examples in This Manual on page xxxix
- Documentation Conventions on page xl
- Documentation Feedback on page xlii
- Requesting Technical Support on page xlii

JUNOS Documentation and Release Notes

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

If the information in the latest release notes differs from the information in the documentation, follow the *JUNOS Software Release Notes*.

To obtain the most current version of all Juniper Networks® technical documentation, see the product documentation page on the Juniper Networks website at http://www.juniper.net/techpubs/.

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

Objectives

This guide is designed for network administrators who are configuring and monitoring a Juniper Networks J Series, M Series, MX Series, or T Series routing platform.

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

Audience

This guide is designed for network administrators who are configuring and monitoring a Juniper Networks M Series, MX Series, T Series, EX Series, or J Series router or switch.

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

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

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

Supported Platforms

For the features described in this manual, JUNOS Software currently supports the following platforms:

- J Series
- M Series

- MX Series
- T Series
- EX Series

Using the Indexes

This reference contains two indexes: a complete index that includes topic entries, and an index of statements and commands only.

In the index of statements and commands, an entry refers to a statement summary section only. In the complete index, the entry for a configuration statement or command contains at least two parts:

- The primary entry refers to the statement summary section.
- The secondary entry, *usage guidelines,* refers to the section in a configuration guidelines chapter that describes how to use the statement or command.

Using the Examples in This Manual

If you want to use the examples in this manual, you can use the load merge or the load merge relative command. These commands cause the software to merge the incoming configuration into the current candidate configuration. If the example configuration contains the top level of the hierarchy (or multiple hierarchies), the example is a *full example*. In this case, use the load merge command.

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

Merging a Full Example

To merge a full example, follow these steps:

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

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

```
system {
    scripts {
        commit {
            file ex-script.xsl;
        }
    }
}
interfaces {
    fxp0 {
```

```
        disable;
        unit 0 {
            family inet {
                address 10.0.0.1/24;
            }
        }
    }
}
```

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

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

Merging a Snippet

To merge a snippet, follow these steps:

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

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

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

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

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

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

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

For more information about the load command, see the *JUNOS CLI User Guide*.

Documentation Conventions

Table 1 on page xli defines notice icons used in this guide.

Table 1: Notice Icons

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

Table 2 on page xli defines the text and syntax conventions used in this guide.

Table 2: Text and Syntax Conventions

Convention	Description	Examples
Bold text like this	Represents text that you type.	To enter configuration mode, type the **configure** command: user@host> **configure**
`Fixed-width text like this`	Represents output that appears on the terminal screen.	`user@host> show chassis alarms` `No alarms currently active`
Italic text like this	■ Introduces important new terms. ■ Identifies book names. ■ Identifies RFC and Internet draft titles.	■ A policy *term* is a named structure that defines match conditions and actions. ■ *JUNOS System Basics Configuration Guide* ■ RFC 1997, *BGP Communities Attribute*
Italic text like this	Represents variables (options for which you substitute a value) in commands or configuration statements.	Configure the machine's domain name: [edit] root@# **set system domain-name** *domain-name*
Plain text like this	Represents names of configuration statements, commands, files, and directories; IP addresses; configuration hierarchy levels; or labels on routing platform components.	■ To configure a stub area, include the stub statement at the [edit protocols ospf area area-id] hierarchy level. ■ The console port is labeled CONSOLE.
< > (angle brackets)	Enclose optional keywords or variables.	stub <default-metric *metric*>;

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

Convention	Description	Examples
| (pipe symbol)	Indicates a choice between the mutually exclusive keywords or variables on either side of the symbol. The set of choices is often enclosed in parentheses for clarity.	broadcast | multicast (*string1* | *string2* | *string3*)
# (pound sign)	Indicates a comment specified on the same line as the configuration statement to which it applies.	rsvp { # Required for dynamic MPLS only
[ ] (square brackets)	Enclose a variable for which you can substitute one or more values.	community name members [*community-ids*]

Indention and braces ({ })	Identify a level in the configuration hierarchy.	[edit]
		routing-options {
		  static {
		  route default {
		  nexthop *address*;
		  retain;
		  }
		  }
		}

; (semicolon)	Identifies a leaf statement at a configuration hierarchy level.	
J-Web GUI Conventions		
Bold text like this	Represents J-Web graphical user interface (GUI) items you click or select.	■ In the Logical Interfaces box, select **All Interfaces**. ■ To cancel the configuration, click **Cancel**.
> (bold right angle bracket)	Separates levels in a hierarchy of J-Web selections.	In the configuration editor hierarchy, select **Protocols > Ospf**.

Documentation Feedback

We encourage you to provide feedback, comments, and suggestions so that we can improve the documentation. You can send your comments to techpubs-comments@juniper.net, or fill out the documentation feedback form at https://www.juniper.net/cgi-bin/docbugreport/. If you are using e-mail, be sure to include the following information with your comments:

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

Requesting Technical Support

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

contract, or are covered under warranty, and need postsales technical support, you can access our tools and resources online or open a case with JTAC.

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

Self-Help Online Tools and Resources

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

- Find CSC offerings: http://www.juniper.net/customers/support/
- Search for known bugs: http://www2.juniper.net/kb/
- Find product documentation: http://www.juniper.net/techpubs/
- Find solutions and answer questions using our Knowledge Base: http://kb.juniper.net/
- Download the latest versions of software and review release notes: http://www.juniper.net/customers/csc/software/
- Search technical bulletins for relevant hardware and software notifications: https://www.juniper.net/alerts/
- Join and participate in the Juniper Networks Community Forum: http://www.juniper.net/company/communities/
- Open a case online in the CSC Case Management tool: http://www.juniper.net/cm/

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

Opening a Case with JTAC

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

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

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

Part 1
Overview

- Routing Protocols Concepts on page 3
- Complete Routing and Routing Protocol Configuration Statements on page 15

Chapter 1

Routing Protocols Concepts

This chapter discusses the following topics:

- Routing Databases Overview on page 3
- Route Preferences Overview on page 6
- How the Active Route Is Determined on page 7
- Default Route Preference Values on page 8
- Equal-Cost Paths and Load Sharing on page 10
- IPv6 Overview on page 10
- IPv6 Standards on page 13

Routing Databases Overview

The JUNOS Software maintains two databases for routing information:

- Routing table—Contains all the routing information learned by all routing protocols.
- Forwarding table—Contains the routes actually used to forward packets through the router.

In addition, the interior gateway protocols (IGPs), IS-IS, and OSPF maintain link-state databases.

This section includes the following topics:

- Routing Protocol Databases on page 3
- JUNOS Routing Tables on page 4
- Forwarding Tables on page 5
- How the Routing and Forwarding Tables Are Synchronized on page 5

Routing Protocol Databases

Each IGP routing protocol maintains a database of the routing information it has learned from other routers running the same protocol and uses this information as defined and required by the protocol. IS-IS and OSPF use the routing information they received to maintain link-state databases, which they use to determine which adjacent neighbors are operational and to construct network topology maps.

IS-IS and OSPF use the Dijkstra algorithm, and RIP and RIPng use the Bellman-Ford algorithm to determine the best route or routes (if there are multiple equal-cost routes) to reach each destination and install these routes into the JUNOS Software routing table.

When you configure a protocol on an interface, you must also configure a protocol family on that interface.

JUNOS Routing Tables

The JUNOS Software routing table is used by the routing protocol process to maintain its database of routing information. In this table, the routing protocol process stores statically configured routes, directly connected interfaces (also called *direct routes* or *interface routes*), and all routing information learned from all routing protocols. The routing protocol process uses this collected routing information to select the *active route* to each destination, which is the route that actually is used to forward packets to that destination.

By default, the JUNOS Software maintains three routing tables: one for unicast routes, another for multicast routes, and a third for MPLS. You can configure additional routing tables to support situations where you need to separate a particular group of routes or where you need greater flexibility in manipulating routing information. In general, most operations can be performed without resorting to the complexity of additional routing tables. However, creating additional routing tables has several specific uses, including importing interface routes into more than one routing table, applying different routing policies when exporting the same route to different peers, and providing greater flexibility with incongruent multicast topologies.

Each routing table is identified by a name, which consists of the protocol family followed by a period and a small, nonnegative integer. The protocol family can be inet (Internet), iso (ISO), or mpls (MPLS). The following names are reserved for the default routing tables maintained by the JUNOS Software:

- inet.0—Default IP version 4 (IPv4) unicast routing table
- inet6.0—Default IP version 6 (IPv6) unicast routing table
- instance-name.inet.0—Unicast routing table for a particular routing instance
- inet.1—Multicast forwarding cache
- inet.2—Unicast routes used for multicast reverse path forwarding (RPF) lookup
- inet.3—MPLS routing table for path information
- mpls.0—MPLS routing table for label-switched path (LSP) next hops

NOTE: For clarity, this manual contains general discussions of routing tables as if there were only one table. However, when it is necessary to distinguish among the routing tables, their names are explicitly used.

Forwarding Tables

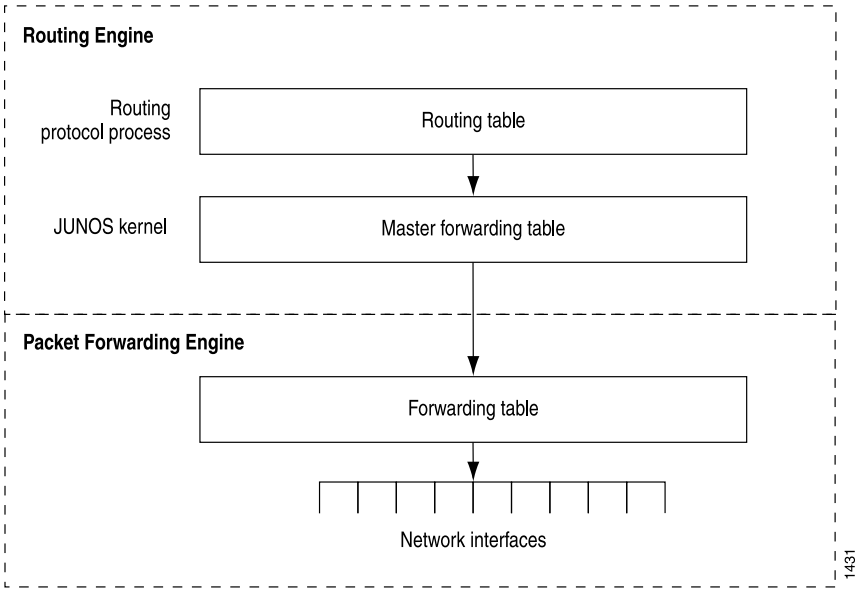
The JUNOS Software installs all active routes from the routing table into the forwarding table. The active routes are used to forward packets to their destinations.

The JUNOS kernel maintains a master copy of the forwarding table. It copies the forwarding table to the Packet Forwarding Engine, which is the part of the router responsible for forwarding packets.

How the Routing and Forwarding Tables Are Synchronized

The JUNOS routing protocol process is responsible for synchronizing the routing information between the routing and forwarding tables. To do this, the routing protocol process calculates the active routes from all the routes in the routing table and installs them into the forwarding table. The routing protocol process then copies the forwarding table to the router's Packet Forwarding Engine, the part of the router that forwards packets. Figure 1 on page 5 illustrates how the routing tables are synchronized.

Figure 1: Synchronizing Routing Exchange Between the Routing and Forwarding Tables



Route Preferences Overview

For unicast routes, the JUNOS routing protocol process uses the information in its routing table, along with the properties set in the configuration file, to choose an *active route* for each destination. While the JUNOS Software might know of many routes to a destination, the active route is the preferred route to that destination and is the one that is installed in the forwarding table and used when actually routing packets.

The routing protocol process generally determines the active route by selecting the route with the lowest preference value. The preference value is an arbitrary value in the range from 0 through 4,294,967,295 ($2^{32} - 1$) that the software uses to rank routes received from different protocols, interfaces, or remote systems.

The preference value is used to select routes to destinations in external autonomous systems (ASs) or routing domains; it has no effect on the selection of routes within an AS (that is, within an interior gateway protocol [IGP]). Routes within an AS are selected by the IGP and are based on that protocol's metric or cost value.

This section includes the following topics:

- Alternate and Tiebreaker Preferences on page 6
- Multiple Active Routes on page 6

Alternate and Tiebreaker Preferences

The JUNOS Software provides support for alternate and tiebreaker preferences, and some of the routing protocols, including BGP and label switching, use these additional preferences. With these protocols, you can specify a primary route preference (by including the preference statement in the configuration), and a secondary preference that is used as a tiebreaker (by including the preference2 statement). You can also mark route preferences with additional route tiebreaker information by specifying a color and a tiebreaker color (by including the color and color2 statements in the configuration). For configuration instructions, see "Configuring a Preference Value for Static Routes" on page 68, "Configuring a Preference Value for Aggregate Routes" on page 90, and "Configuring a Preference Value for Generated Routes" on page 99.

The software uses a 4-byte value to represent the route preference value. When using the preference value to select an active route, the software first compares the primary route preference values, choosing the route with the lowest value. If there is a tie and a secondary preference has been configured, the software compares the secondary preference values, choosing the route with the lowest value. The secondary preference values must be included in a set for the preference values to be considered.

Multiple Active Routes

The IGPs compute equal-cost multipath next hops, and IBGP picks up these next hops. When there are multiple, equal-cost next hops associated with a route, the routing protocol process installs only one of the next hops in the forwarding path with each route, randomly selecting which next hop to install. For example, if there are 3 equal-cost paths to an exit routing device and 900 routes leaving through that

routing device, each path ends up with about 300 routes pointing at it. This mechanism provides load distribution among the paths while maintaining packet ordering per destination.

How the Active Route Is Determined

For each prefix in the routing table, the routing protocol process selects a single best path, called the active route. The algorithm for determining the active route is as follows:

1. Choose the path with the lowest preference value (routing protocol process preference). Routes that are not eligible to be used for forwarding (for example, because they were rejected by routing policy or because a next hop is inaccessible) have a preference of –1 and are never chosen.
2. For BGP, prefer the path with higher local preference. For non-BGP paths, choose the path with the lowest preference2 value.
3. If the path includes an AS path:
   a. Prefer the route with a shorter AS path.

      Confederation sequences are considered to have a path length of 0, and AS and confederation sets are considered to have a path length of 1.

   b. Prefer the route with the lower origin code. Routes learned from an IGP have a lower origin code than those learned from an EGP, and both have lower origin codes than incomplete routes (routes whose origin is unknown).

   c. Depending on whether nondeterministic routing table path selection behavior is configured, there are two possible cases:

      - If nondeterministic routing table path selection behavior is not configured (that is, if the path-selection cisco-nondeterministic statement is not included in the BGP configuration), for paths with the same neighboring AS numbers at the front of the AS path, prefer the path with the lowest multiple exit discriminator (MED) metric. Confederation AS numbers are not considered when deciding what the neighbor AS number is. When you display the routes in the routing table using the show route command, they generally appear in order from most preferred to least preferred. Routes that share the same neighbor AS are grouped together in the command output. Within a group, the best route is listed first and the other routes are marked with the NotBest flag in the State field of the show route detail command.
      - To always compare MEDs whether or not the peer ASs of the compared routes are the same, include the path-selection always-compare-med statement. For an example, see “Configuring Routing Table Path Selection for BGP” on page 754.

      If nondeterministic routing table path selection behavior is configured (that is, the path-selection cisco-nondeterministic statement is included in the BGP configuration), prefer the path with the lowest MED metric. When you display the routes in the routing table using the show route command, they generally appear in order from most preferred to least preferred and are ordered with

the best route first, followed by all other routes in order from newest to oldest.

In both cases, confederations are not considered when determining neighboring ASs. Also, in both cases, a missing metric is treated as if a MED were present but zero.

4. Prefer strictly internal paths, which include IGP routes and locally generated routes (static, direct, local, and so forth).
5. Prefer strictly external (EBGP) paths over external paths learned through interior sessions (IBGP).
6. For BGP, prefer the path whose next hop is resolved through the IGP route with the lowest metric.
7. For BGP, if both paths are external, prefer the currently active path to minimize route-flapping. This rule is not used if:
   a. path-selection external-router-id is configured.
   b. Both peers have the same router-id.
   c. Either peer is a confederation peer.
   d. Neither path is the current active path.
8. For BGP, prefer the path from the peer with the lowest router ID; for any path with an originator ID attribute, substitute the originator ID for the router ID during router ID comparison.
9. For BGP, prefer the path with the shortest cluster list length; length is 0 for no list.
10. For BGP, prefer the path from the peer with the lowest peer IP address.

Default Route Preference Values

The JUNOS Software routing protocol process assigns a default preference value to each route that the routing table receives. The default value depends on the source of the route. The preference value is a value from 0 through 4,294,967,295 ($2^{32} - 1$), with a lower value indicating a more preferred route. Table 3 on page 8 lists the default preference values.

Table 3: Default Route Preference Values

How Route Is Learned	Default Preference	Statement to Modify Default Preference
Directly connected network	0	–
System routes	4	–
Static and Static LSPs	5	static

Table 3: Default Route Preference Values *(continued)*

How Route Is Learned	Default Preference	Statement to Modify Default Preference
RSVP-signaled LSPs	7	RSVP preference as described in the *JUNOS MPLS Applications Configuration Guide*
LDP-signaled LSPs	9	LDP preference, as described in the *JUNOS MPLS Applications Configuration Guide*
OSPF internal route	10	OSPF preference
IS-IS Level 1 internal route	15	IS-IS preference
IS-IS Level 2 internal route	18	IS-IS , preference
Redirects	30	–
Kernel	40	–
SNMP	50	–
Router discovery	55	–
RIP	100	RIP preference
RIPng	100	RIPng preference
PIM	105	*JUNOS Multicast Protocols Configuration Guide*
DVMRP	110	*JUNOS Multicast Protocols Configuration Guide*
Aggregate	130	aggregate
OSPF AS external routes	150	OSPF external-preference
IS-IS Level 1 external route	160	IS-IS external-preference
IS-IS Level 2 external route	165	IS-IS external-preference
BGP	170	BGP preference, export, import
MSDP	175	*JUNOS Multicast Protocols Configuration Guide*

In general, the narrower the scope of the statement, the higher precedence its preference value is given, but the smaller the set of routes it affects. To modify the default preference value for routes learned by routing protocols, you generally apply routing policy when configuring the individual routing protocols. You also can modify some preferences with other configuration statements, which are indicated in the table. For information about defining and applying routing policies, see the *JUNOS Policy Framework Configuration Guide*.

Equal-Cost Paths and Load Sharing

For equal-cost paths, load sharing is based on the BGP next hop. For example, if four prefixes all point to a next hop and there is more than one equal-cost path to that next hop, the routing protocol process uses a hash algorithm to choose the path among the four prefixes. Also, for each prefix, the routing protocol process installs a single forwarding entry pointing along one of the paths. The routing software does not rehash the path taken as prefixes pointing to the next hop come and go, but it does rehash if the number of paths to the next hop changes. Because a prefix is tied to a particular path, packet reordering should not happen. The degree of load sharing improves as the number of prefixes increases.

IPv6 Overview

IP version 6 (IPv6) is the latest version of IP. IP enables numerous nodes on different networks to interoperate seamlessly. IP version 4 (IPv4) is currently used in intranets and private networks, as well as the Internet. IPv6 is the successor to IPv4, and is based for the most part on IPv4.

IPv4 has been widely deployed and used to network the Internet today. With the rapid growth of the Internet, enhancements to IPv4 are needed to support the influx of new subscribers, Internet-enabled devices, and applications. IPv6 is designed to enable the global expansion of the Internet.

IPv6 builds upon the functionality of IPv4, providing improvements to addressing, configuration and maintenance, and security.

IPv6 offers the following benefits:

- Expanded addressing capabilities—IPv6 provides a larger address space. IPv6 addresses consist of 128 bits, while IPv4 addresses consist of 32 bits. 128-bit addressing increases the address space by approximately 1029 unique addresses, enough to last for the forseeable future.
- Header format simplification—IPv6 packet header format is designed to be efficient. IPv6 standardizes the size of the packet header to 40 bytes, divided into 8 fields.
- Improved support for extensions and options—Extension headers carry Internet-layer information and have a standard size and structure.
- Flow labeling capability—Flow labels provide consistent handling of packets belonging to the same flow.
- Improved privacy and security—IPv6 supports extensions for authentication and data integrity, which enhance privacy and security.

This section discusses the following topics:

- IPv6 Packet Headers on page 11
- IPv6 Addressing on page 12

IPv6 Packet Headers

IPv6 headers are different from IPv4 headers.

This section discusses the following topics that provide background information about IPv6 headers:

- Header Structure on page 11
- Extension Headers on page 11

Header Structure

IPv6 packet headers contain many of the fields found in IPv4 packet headers; some of these fields have been modified from IPv4. The 40-byte IPv6 header consists of the following 8 fields:

- Traffic class—Class-of-service (CoS) priority of the packet. Previously the type-of-service (ToS) field in IPv4. However, the semantics of this field (for example, DiffServ code points) are identical to IPv4.
- Destination address—Final destination node address for the packet.
- Flow label—Packet flows requiring a specific class of service. The flow label identifies all packets belonging to a specific flow, and routers can identify these packets and handle them in a similar fashion.
- Hop limit—Maximum number of hops allowed. Previously the time-to-live (TTL) field in IPv4.
- Next header—Next extension header to examine. Previously the protocol field in IPv4.
- Payload length—Length of the IPv6 payload. Previously the total length field in IPv4.
- Source address—Address of the source node sending the packet.
- Version—Version of IP.

Extension Headers

In IPv6, *extension headers* are used to encode optional Internet-layer information.

Extension headers are placed between the IPv6 header and the upper layer header in a packet.

Extension headers are chained together using the next header field in the IPv6 header. The next header field indicates to the router which extension header to expect next. If there are no more extension headers, the next header field indicates the upper layer header (TCP header, User Datagram Protocol [UDP] header, ICMPv6 header, an encapsulated IP packet, or other items).

IPv6 Addressing

IPv6 uses a 128-bit addressing model. This creates a much larger address space than IPv4 addresses, which are made up of 32 bits. IPv6 addresses also contain a scope field that categorizes what types of applications are suitable for the address. IPv6 does not support broadcast addresses, but instead uses multicast addresses to serve this role. In addition, IPv6 also defines a new type of address called *anycast*.

This section discusses the following topics that provide background information about IPv6 addressing:

- Address Representation on page 12
- Address Types on page 12
- Address Scope on page 13
- Address Structure on page 13

Address Representation

IPv6 addresses consist of 8 groups of 16-bit hexadecimal values separated by colons (:). The IPv6 address format is as follows:

aaaa:*aaaa*:*aaaa*:*aaaa*:*aaaa*:*aaaa*:*aaaa*:*aaaa*

aaaa is a 16-bit hexadecimal value, and *a* is a 4-bit hexadecimal value. Following is an example of an actual IPv6 address:

3FFE:0000:0000:0001:0200:F8FF:FE75:50DF

You can omit the leading zeros, as shown:

3FFE:0:0:1:200:F8FF:FE75:50DF

You can compress 16-bit groups of zeros to the notation :: (two colons), as shown here, but only once per address:

3FFE::1:200:F8FF:FE75:50DF

Address Types

There are three types of IPv6 addresses:

- Unicast—For a single interface.
- Multicast—For a set of interfaces on the same physical medium. A packet is sent to all of the interfaces associated with the address.
- Anycast—For a set of interfaces on different physical mediums. A packet is sent to only one of the interfaces associated with this address, not to all the interfaces.

Address Scope

IPv6 addresses have *scope*, which identifies the application suitable for the address. Unicast and multicast addresses support scoping.

Unicast addresses support two types of scope: *global* scope and *local* scope. There are two types of local scope: *link-local* addresses and *site-local* addresses. Link-local unicast addresses are used within a single network link. The first ten bits of the prefix identify the address as a link-local address. Link-local addresses cannot be used outside a network link. Site-local unicast addresses are used within a site or intranet. A site consists of multiple network links, and site-local addresses identify nodes inside the intranet. Site-local addresses cannot be used outside the site.

Multicast addresses support 16 different types of scope, including node, link, site, organization, and global scope. A 4-bit field in the prefix identifies the scope.

Address Structure

Unicast addresses identify a single interface. The address consists of *n* bits for the prefix, and 128 – *n* bits for the interface ID.

Multicast addresses identify a set of interfaces. The address is made up of the first 8 bits of all ones, a 4-bit flags field, a 4-bit scope field, and a 112-bit group ID:

11111111 | *flags* | *scope* | *group ID*

The first octet of ones identifies the address as a multicast address. The flags field identifies whether the multicast address is a well-known address or a transient multicast address. The scope field identifies the scope of the multicast address. The 112-bit group ID identifies the multicast group.

Similar to multicast addresses, anycast addresses identify a set of interfaces. However, packets are sent to only one of the interfaces, not to all interfaces. Anycast addresses are allocated from the normal unicast address space and cannot be distinguished from a unicast address in format. Therefore, each member of an anycast group must be configured to recognize certain addresses as anycast addresses.

IPv6 Standards

IPv6 is defined in the following documents:

- RFC 1981, *Path MTU Discovery for IP version 6*
- RFC 2373, *IP Version 6 Addressing Architecture*
- RFC 2460, *Internet Protocol, Version 6 (IPv6)*
- RFC 2461, *Neighbor Discovery for IP Version 6*
- RFC 2462, *IPv6 Stateless Address Auto configuration*
- RFC 2463, *Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6*

- RFC 2464, *Transmission of IPv6 Packets over Ethernet Networks*
- RFC 2472, *IP Version 6 over PPP*
- RFC 2474, *Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers*
- RFC 2675, *IPv6 Jumbo grams*
- RFC 2767, *Dual Stack Hosts using the "Bump-In-the-Stack" Technique (BIS)*
- RFC 2878, *PPP Bridging Control Protocol*
- RFC 2893, *Transition Mechanisms for IPv6 Hosts and Routers*
- Internet draft draft-ietf-dhc-dhcpv6-16.txt, *Dynamic Host Configuration Protocol for IPv6* (expires May 2001)
- Internet draft draft-kato-bgp-ipv6-link-local-00.txt, *BGP4 + Peering Using IPv6 Link-local Address* (expires April 2002)
- Internet draft draft-ietf-idr-flow-spec-00.txt, *Dissemination of Flow Specification Rules*

To access Internet Requests for Comments (RFCs) and drafts, see http://www.ietf.org.

Chapter 2

Complete Routing and Routing Protocol Configuration Statements

For a list of the complete configuration statement hierarchy, see the *JUNOS Hierarchy and RFC Reference*.

This chapter is organized as follows:

- [edit logical-systems] Hierarchy Level on page 15
- [edit protocols] Hierarchy Level on page 16
- [edit routing-instances] Hierarchy Level on page 33
- [edit routing-options] Hierarchy Level on page 38

[edit logical-systems] Hierarchy Level

The following lists the statements that can be included at the [edit logical-systems] hierarchy level and are also documented in this manual.

```
logical-systems {
    logical-system-name {
        protocols {
            bgp {
                bgp-configuration;
            }
            isis {
                isis-configuration;
            }
            ospf {
                ospf-configuration;
            }
            ospf3 {
                ospf3-configuration;
            }
            rip {
                rip-configuration;
            }
            ripng {
                ripng-configuration;
            }
            router-advertisement {
                router-advertisement-configuration;
```

```
        }
        router-discovery {
            router-discovery-configuration;
        }
    }
    routing-instances {
        routing-instance-name {
            routing-instance-configuration;
        }
    }
    routing-options {
        routing-option-configuration;
    }
  }
}
```

[edit protocols] Hierarchy Level

The following statements can also be included at the [edit logical-systems *logical-system-name*] hierarchy level.

```
protocols {
```

BGP Global

```
bgp {
    accept-remote-nexthop;
    advertise-external <conditional>;
    advertise-inactive;
    (advertise-peer-as | noadvertise-peer-as);
    authentication-algorithm algorithm;
    authentication-key key;
    authentication-key-chain key-chain;
    bfd-liveness-detection{
        authentication {
            algorithm algorithm-name;
            key-chain key-chain-name;
            loose-check;
        }
        detection-time {
            threshold milliseconds;
        }
        holddown-interval milliseconds;
        minimum-interval milliseconds;
        minimum-receive-interval milliseconds;
        no-adaptation;
        transmit-interval {
            threshold milliseconds;
            minimum-interval milliseconds;
        }
        multiplier number;
        version (1 | automatic);
    }
    cluster cluster-identifier;
    damping;
    description text-description;
    disable;
```

```
export [ policy-names ];
family family-name{
... family-configuration ...
graceful-restart {
   disable;
   restart-time seconds;
   stale-routes-time seconds;
}
group group-name {
... group-configuration ...
}
hold-time seconds;
idle-after-switch-over (seconds | forever);
import [ policy-names ];
include-mp-next-hop;
ipsec-sa ipsec-sa;
keep (all | none);
local-address address;
local-as autonomous-system <private>;
local-interface interface-name;
local-preference local-preference;
log-updown;
metric-out (metric | igp (delay-med-update | <metric-offset>) | minimum-igp
   <metric-offset>);
mtu-discovery;
multihop {
   no-nexthop-change;
   ttl ttl-value;
}
no-aggregator-id;
no-client-reflect;
outbound-route-filter{
   bgp-orf-cisco-mode;
   prefix-based {
      accept {
         (inet | inet6);
      }
   }
}
out-delay seconds;
passive;
path-selection {
   (cisco-non-deterministic | always-compare-med | external-router-id);
   med-plus-igp {
      igp-multiplier number;
      med-multiplier number;
   }
}
peer-as autonomous-system;
preference preference;
remove-private;
tcp-mss segment-size;
traceoptions {
   file filename <files number> <size size> <world-readable | no-world-readable>;
   flag flag <flag-modifier> <disable>;
}
```

```
    vpn-apply-export;
}
```

BGP Family

```
family {
    (inet | inet6 | inet-vpn | inet6-vpn | iso-vpn) {
        (any | flow | labeled-unicast | multicast | unicast) {
            accepted-prefix-limit {
                maximum number;
                teardown <percentage> <idle-timeout (forever | minutes)>;
            }
            <loops number>;
            prefix-limit {
                maximum number;
                teardown <percentage> <idle-timeout (forever | minutes)>;
            }
            rib-group group-name;
        }
        flow{
            no-validate policy-name;
        }
        labeled-unicast {
            accepted-prefix-limit {
                maximum number;
                teardown <percentage> <idle-timeout (forever | minutes)>;
            }
            aggregate-label {
                community community-name:
            }
            explicit-null {
                connected-only;
            }
            prefix-limit {
                maximum number;
                teardown <percentage> <idle-timeout (forever | minutes)>;
            }
            resolve-vpn;
            rib inet.3;
            rib-group group-name;
        }
    }
    route-target {
        accepted-prefix-limit {
            maximum number;
            teardown <percentage> <idle-timeout (forever | minutes)>;
        }
        advertise-default;
        external-paths number;
        prefix-limit {
            maximum number;
            teardown <percentage> <idle-timeout (forever | minutes)>;
        }
    }
    (inet-mdt | inet-mvpn | inet6-mvpn | l2-vpn) {
        signaling {
            accepted-prefix-limit {
```

```
            maximum number;
            teardown <percentage> <idle-timeout (forever | minutes)>;
          }
          <loops number>;
          prefix-limit {
            maximum number;
            teardown <percentage> <idle-timeout (forever | minutes)>;
            }
          rib-group group-name
        }
      }
    }
```

BGP Group

```
group group-name {
  accept-remote-nexthop;
  advertise-external <conditional>;
  advertise-inactive;
  advertise-peer-as;
  allow ([ network/mask-length ] | all);
  as-override;
  authentication-algorithm algorithm;
  authentication-key key;
  authentication-key-chain key-chain;
  bfd-liveness-detection {
    authentication {
      algorithm algorithm-name;
      key-chain key-chain-name;
      loose-check;
    }
    detection-time {
      threshold milliseconds;
    }
    holddown-interval milliseconds;
    minimum-interval milliseconds;
    minimum-receive-interval milliseconds;
    multiplier number;
    no-adaptation;
    transmit-interval {
      threshold milliseconds;
      minimum-interval milliseconds;
    }
    version (1 | automatic);
  }
  cluster cluster-identifier;
  damping;
  description text-description;
  export [ policy-names ];
  family {
    (inet | inet6 | inet-vpn | inet6-vpn | iso-vpn) {
      (any | flow | labeled-unicast | multicast | unicast) {
        accepted-prefix-limit {
          maximum number;
          teardown <percentage> <idle-timeout (forever | minutes)>;
        }
        <loops number>;
```

```
            prefix-limit {
                maximum number;
                teardown <percentage> <idle-timeout (forever | minutes)>;
            }
            rib-group group-name;
        }
        flow {
            no-validate policy-name;
        }
        labeled-unicast {
            accepted-prefix-limit {
                maximum number;
                teardown <percentage> <idle-timeout (forever | minutes)>;
            }
            aggregate-label {
                community community-name:
            }
            explicit-null {
                connected-only;
            }
            prefix-limit {
                maximum number;
                teardown <percentage> <idle-timeout (forever | minutes)>;
            }
            resolve-vpn;
            rib inet.3;
            rib-group group-name;
        }
    }
    route-target {
        accepted-prefix-limit {
            maximum number;
            teardown <percentage> <idle-timeout (forever | minutes)>;
        }
        advertise-default;
        external-paths number;
        prefix-limit {
            maximum number;
            teardown <percentage> <idle-timeout (forever | minutes)>;
        }
    }
    (inet-mdt | inet-mvpn | inet6-mvpn | l2-vpn) {
        signaling {
            accepted-prefix-limit {
                maximum number;
                teardown <percentage> <idle-timeout (forever | minutes)>;
            }
            <loops number>;
            prefix-limit {
                maximum number;
                teardown <percentage> <idle-timeout (forever | minutes)>;
                }
            rib-groupgroup-name
        }
    }
}
```

```
    graceful-restart {
        disable;
        restart-time seconds;
        stale-routes-time seconds;
    }
    hold-time seconds;
    idle-after-switch-over (seconds | forever);
    import [ policy-names ];
    include-mp-next-hop;
    ipsec-sa ipsec-sa;
    keep (all | none);
    local-address address;
    local-as autonomous-system <private>;
    local-interface;
    local-preference local-preference;
    log-updown;
    metric-out (metric | minimum-igp <offset> | igp <offset>);
    mtu-discovery;
    multihop <ttl-value>;
    multipath {
        multiple-as;
    }
    no-advertise-peer-as;
    no-aggregator-id;
    no-client-reflect;
    outbound-route-filter {
        bgp-orf-cisco-mode;
        prefix-based {
            accept {
                (inet | inet6);
            }
        }
    }
    out-delay seconds;
    passive;
    peer-as autonomous-system;
    preference preference;
    remove-private;
    tcp-mss segment-size;
    traceoptions {
        file filename <files number> <size size> <world-readable | no-world-readable>;
        flag flag <flag-modifier> <disable>;
    }
    type type;
vpn-apply-export;
```

BGP Neighbor

```
neighbor address {
    accept-remote-nexthop;
    advertise-external <conditional>;
    advertise-inactive;
    advertise-peer-as;
    as-override;
    authentication-algorithm algorithm;
    authentication-key key;
    authentication-key-chain key-chain;
```

```
bfd-liveness-detection {
   authentication {
      algorithm algorithm-name;
      key-chain key-chain-name;
      loose-check;
   }
   detection-time {
      threshold milliseconds;
   }
   minimum-interval milliseconds;
   minimum-receive-interval milliseconds;
   multiplier number;
   no-adaptation;
   transmit-interval {
      threshold milliseconds;
      minimum-interval milliseconds;
   }
   version (1 | automatic);
}
cluster cluster-identifier;
damping;
description text-description;
export [ policy-names ];
family{
   (inet | inet6 | inet-vpn | inet6-vpn | iso-vpn) {
      (any | flow | labeled-unicast | multicast | unicast) {
         accepted-prefix-limit {
            maximum number;
            teardown <percentage> <idle-timeout (forever | minutes)>;
         }
         <loops number>;
         prefix-limit {
            maximum number;
            teardown <percentage> <idle-timeout (forever | minutes)>;
         }
         rib-group group-name;
      }
      flow {
         no-validate policy-name;
      }
      labeled-unicast {
         accepted-prefix-limit {
            maximum number;
            teardown <percentage> <idle-timeout (forever | minutes)>;
         }
         aggregate-label {
            community community-name:
         }
         explicit-null {
            connected-only;
         }
         prefix-limit {
            maximum number;
            teardown <percentage> <idle-timeout (forever | minutes)>;
         }
         resolve-vpn;
```

```
            rib inet.3;
            rib-group group-name;
        }
    }
    route-target {
        accepted-prefix-limit {
            maximum number;
            teardown <percentage> <idle-timeout (forever | minutes)>;
        }
        advertise-default;
        external-paths number;
        prefix-limit {
            maximum number;
            teardown <percentage> <idle-timeout (forever | minutes)>;
        }
    }
    (inet-mdt | inet-mvpn | inet6-mvpn | l2-vpn) {
        signaling {
            accepted-prefix-limit {
                maximum number;
                teardown <percentage> <idle-timeout (forever | minutes)>;
            }
            <loops number>;
            prefix-limit {
                maximum number;
                teardown <percentage> <idle-timeout (forever | minutes)>;
                }
            rib-group group-name
        }
    }
}
graceful-restart {
    disable;
    restart-time seconds;
    stale-routes-time seconds;
}
hold-time seconds;
idle-after-switch-over (seconds | forever);
import [ policy-names ];
include-mp-next-hop;
ipsec-sa ipsec-sa;
keep (all | none);
local-address address;
local-as autonomous-system <private>;
local-interface interface-name;
local-preference local-preference;
log-updown;
metric-out (metric | minimum-igp <offset> | igp <offset>);
mtu-discovery;
multihop <ttl-value>;
multipath {
    multiple-as;
}
no-advertise-peer-as;
no-aggregator-id;
no-client-reflect;
```

```
        out-delay seconds;
        outbound-route-filter {
            bgp-orf-cisco-mode;
            prefix-based {
                accept {
                    (inet | inet6);
                }
            }
        }
        passive;
        peer-as autonomous-system;
        preference preference;
        remove-private;
        tcp-mss segment-size;
        traceoptions {
            file filename <files number> <size size> <world-readable | no-world-readable>;
            flag flag <flag-modifier> <disable>;
        }
    vpn-apply-export;
```

IS-IS

```
isis {
    clns-routing;
    disable;
    export [ policy-names ];
    graceful-restart {
        disable;
        helper-disable;
        restart-duration seconds;
    }
    ignore-attached-bit;
    interface(all | interface-name) {
        bfd-liveness-detection {
            authentication {
                algorithm algorithm-name;
                key-chain key-chain-name;
                loose-check;
            }
            detection-time {
                threshold milliseconds;
            }
            minimum-interval milliseconds;
            minimum-receive-interval milliseconds;
            transmit-interval {
                threshold milliseconds;
                minimum-interval milliseconds;
            }
            multiplier number;
        }
        checksum;
        csnp-interval (seconds | disable);
        disable;
        hello-padding (adaptive | loose | strict);
        ldp-synchronization {
            disable;
            hold-time seconds;
```

```
    }
    level level-number {
        disable;
        hello-authentication-type authentication;
        hello-authentication-key key;
        hello-interval seconds;
        hold-time seconds;
        ipv4-multicast-metric number;
        ipv6-multicast-metric number;
        ipv6-unicast-metric number;
        metric metric;
        passive;
        priority number;
        te-metric metric;
    }
    link-protection;
    lsp-interval milliseconds;
    mesh-group (value | blocked);
    no-adjacency-holddown;
    no-eligible-backup;
    no-ipv4-multicast;
    no-ipv6-multicast;
    no-ipv6-unicast;
    no-unicast-topology;
    node-link-protection;
    passive;
    point-to-point;
}
label-switched-path namelevel level metric metric;
level level-number {
    authentication-key key;
    authentication-type authentication;
    external-preference preference;
    ipv6-multicast-metric number;
    no-csnp-authentication;
    no-hello-authentication;
    no-psnp-authentication;
    preference preference;
    prefix-export-limit number;
    wide-metrics-only;
}
loose-authentication-check;
lsp-lifetime seconds;
max-areas number;
no-adjacency-holddown;
no-authentication-check;
no-ipv4-routing;
no-ipv6-routing;
overload {
    advertise-high-metrics;
    timeout seconds;
}
reference-bandwidth reference-bandwidth;
rib-group {
    inet group–name;
    inet6 group–name;
```

```
        }
        spf-options {
            delay milliseconds;
            holddown milliseconds;
            rapid-runs number;
        }
        topologies {
            ipv4-multicast;
            ipv6-multicast;
            ipv6-unicast;
        }
        traceoptions {
            file filename <files number> <size size> <world-readable | no-world-readable>;
            flag flag <flag-modifier> <disable>;
        }
        traffic-engineering {
            disable;
            family inet {
                shortcuts <ignore-lsp-metrics> {
                    multicast-rpf-routes;
                }
            }
            family inet6 {
                shortcuts;
            }
        }
    }
```

OSPF

```
ospf {
    disable;
    export [ policy-names ];
    external-preference preference;
    graceful-restart {
        disable;
        helper-disable;
        notify-duration seconds;
        restart-duration seconds;
    }
    import [ policy-names ];
    no-nssa-abr;
    overload {
        timeout seconds;
    }
    preference preference;
    reference-bandwidth reference-bandwidth;
    rib-group group-name;
    sham-link {
        local address;
    }
    spf-options {
        delay milliseconds;
        holddown milliseconds;
        rapid-runs milliseconds;
    }
    traffic-engineering {
```

```
    accept-unnumbered-interfaces;
    multicast-rpf-routes;
    no-topology;
    shortcuts {
        ignore-lsp-metrics;
        lsp-metric-into-summary;
    }
}
traceoptions {
    file filename <files number> <size size> <world-readable | no-world-readable>;
    flag flag <flag-modifier> <disable>;
}
area area-id {
    area-range network/mask-length <restrict> <exact> <override-metric metric>;
    interface interface-name {
        demand-circuit;
        disable;
        bfd-liveness-detection {
            authentication {
                algorithm algorithm-name;
                key-chain key-chain-name;
                loose-check;
            }
            detection-time {
                threshold milliseconds;
            }
            minimum-interval milliseconds;
            minimum-receive-interval milliseconds;
            multiplier number;
            no-adaptation;
            transmit-interval {
                threshold milliseconds;
                minimum-interval milliseconds;
            }
            version (1 | automatic);
        }
        ipsec-sa name;
    }
    authentication {
        md5 key-id {
            key [ key-values ];
        }
        simple-password key-id;
    }
    dead-interval seconds;
    hello-interval seconds;
    interface-type type;
    ldp-synchronization {
        disable;
        hold-time seconds;
    }
    metric metric;
    neighbor address <eligible>;
    network-summary-export [ policy-names ];
    network-summary-import [ policy-names ];
    passive;
```

```
        poll-interval seconds;
        priority number;
        retransmit-interval seconds;
        te-metric metric;
        transit-delay seconds;
      }
      label-switched-path name metric metric;
      nssa {
        area-range network/mask-length <restrict> <exact> <override-metric metric>;
        default-lsa {
          default-metric metric;
          metric-type type;
          type-7;
        }
        (no-summaries | summaries);
      }
      peer-interface interface-name {
        disable;
        dead-interval seconds;
        hello-interval seconds;
        retransmit-interval seconds;
        transit-delay seconds;
      }
      sham-link-remote address {
        ipsec-sa name;
      }
      demand-circuit;
      metric metric;
    }
    stub <default-metric metric> <(no-summaries | summaries)>;
    virtual-link neighbor-id router-id transit-area area-id {
      disable;
      ipsec-sa name;
    }
    authentication {
      md5 key-id;
      simple-password key-id;
    }
    dead-interval seconds;
    hello-interval seconds;
    retransmit-interval seconds;
    transit-delay seconds;
```

OSPFv3

```
ospf3 {
  disable;
  export [ policy-names ];
  external-preference preference;
  import [ policy-names ];
  overload {
    timeout seconds;
  }
  preference preference;
  reference-bandwidth reference-bandwidth;
  rib-group group-name;
  spf-options {
```

```
        delay milliseconds;
        holddown milliseconds;
        rapid-runs number;
    }
    traceoptions {
        file filename <files number> <size size> <world-readable | no-world-readable>;
        flag flag <flag-modifier> <disable>;
    }
    area area-id {
        area-range network/mask-length <restrict> <exact> <override-metric metric>;
        interface interface-name {
            disable;
            dead-interval seconds;
            hello-interval seconds;
            ipsec-sa name;
            metric metric;
            neighbor address <eligible>;
            passive;
            priority number;
            retransmit-interval seconds;
            transit-delay seconds;
        }
        inter-area-prefix-export [policy-names ];
        inter-area-prefix-import [ policy-names ];
        nssa {
            area-range network/mask-length <restrict> <exact> <override-metric metric>;
            default-lsa {
                default-metric metric;
                metric-type type;
                type-7;
            }
            (no-summaries | summaries);
        }
        stub <default-metric metric> <(no-summaries | summaries)>;
        virtual-link neighbor-id router-id transit-area area-id {
            disable;
            dead-interval seconds;
            hello-interval seconds;
            ipsec-sa name;
            retransmit-interval seconds;
            transit-delay seconds;
        }
    }
}
```

RIP

```
rip {
    any-sender;
    authentication-key password;
    authentication-type type;
    (check-zero | no-check-zero);
    graceful-restart {
        disable;
        restart-time seconds;
    }
    holddown seconds;
```

```
import [ policy-names ];
message-size number;
metric-in metric;
receive receive-options;
rib-group group-name;
route-timeout seconds;
send send-options;
update-interval seconds;
traceoptions {
   file filename <files number> <size size> <world-readable | no-world-readable>;
   flag flag <flag-modifier> <disable>;
}
group group-name {
   bfd-liveness-detection {
      authentication {
         algorithm algorithm-name;
         key-chain key-chain-name;
         loose-check;
      }
      detection-time {
         threshold milliseconds;
      }
      minimum-interval milliseconds;
      minimum-receive-interval milliseconds;
      multiplier number;
      no-adaptation;
      transmit-interval {
         threshold milliseconds;
         minimum-interval milliseconds;
      }
      version (0 | 1 | automatic);
   }
   export [ policy-names ];
   metric-out metric;
   preference preference;
   route-timeout seconds;
   update-interval seconds;
   neighbor neighbor-name {
      authentication-key password;
      authentication-type type;
      bfd-liveness-detection {
         authentication {
            algorithm algorithm-name;
            key-chain key-chain-name;
            loose-check;
         }
         detection-time {
            threshold milliseconds;
         }
         minimum-interval milliseconds;
         minimum-receive-interval milliseconds;
         transmit-interval {
            threshold milliseconds;
            minimum-interval milliseconds;
         }
```

```
            multiplier number;
            version (1 | automatic);
          }
          (check-zero | no-check-zero);
          import [ policy-names ];
          message-size number;
          metric-in metric;
          receive receive-options;
          route-timeout seconds;
          send send-options;
          update-interval seconds;
        }
      }
    }
```

RIPng

```
ripng {
  graceful-restart {
    disable;
    restart-time seconds;
  }
  holddown seconds;
  import [ policy-names ];
  metric-in metric;
  receive <none>;
  route-timeout seconds;
  send <none>;
  update-interval seconds;
  traceoptions {
    file filename <files number> <size size> <world-readable | no-world-readable>;
    flag flag <flag-modifier> <disable>;
  }
  group group-name {
    export [ policy-names ];
    metric-out metric;
    preference number;
    route-timeout seconds;
    update-interval seconds;
    neighbor neighbor-name {
      import [ policy-names ];
      metric-in metric;
      receive <none>;
      route-timeout seconds;
      send <none>;
      update-interval seconds;
    }
  }
}
```

Router Advertisement

```
router-advertisement {
  interface interface-name {
    current-hop-limit number;
    default-lifetime seconds;
    (managed-configuration | no-managed-configuration);
    max-advertisement-interval seconds;
    min-advertisement-interval seconds;
```

```
    (other-stateful-configuration | no-other-stateful-configuration);
    prefix prefix {
      (autonomous | no-autonomous);
      (on-link | no-on-link);
      preferred-lifetime seconds;
      valid-lifetime seconds;
    }
    reachable-time milliseconds;
    retransmit-timer milliseconds;
    traceoptions {
      file filename <files number> <size size> <world-readable | no-world-readable>;
      flag flag <detail> <disable>;
    }
  }
}
```

Router Discovery

```
router-discovery {
  disable;
  traceoptions {
    file filename <files number> <size size> <world-readable | no-world-readable>;
    flag flag <detail> <disable>;
  }
  interface interface-name {
    min-advertisement-interval seconds;
    max-advertisement-interval seconds;
    lifetime seconds;
  }
  address address {
    (advertise | ignore);
    (broadcast | multicast);
    (priority number | ineligible);
  }
}
```

Secure Neighbor Discovery

```
neighbor-discovery {
  secure {
    security-level {
      (default | secure-messages-only);
    }
    cryptographic-address {
      key-length number;
      key-pair pathname;
    }
    timestamp {
      clock-drift number;
      known-peer-window seconds;
      new-peer-window seconds;
    }
    traceoptions {
      file filename <files number> <match regular-expression> <size size>
        <world-readable | no-world-readable>;
      flag flag;
      no-remote-trace;
    }
  }
```

```
}
```

[edit routing-instances] Hierarchy Level

```
routing-instances {
   routing-instance-name {
      bridge-domains bridge-domain-name {
         domain-type bridge;
         <vlan-id (all | none | number)>;
         <vlan-tags outer number inner number>;
         <routing-interface routing-interface-name>;
         interface interface-name;
         bridge-options {
            interface-mac-limit limit;
            mac-statistics;
            mac-table-size limit;
            no-mac-learning;
            static-mac mac-address;
         }
      }
      description text;
      forwarding-options;
      interface interface-name;
      instance-type (forwarding | layer2–control | l2vpn | no-forwarding | virtual-router |
         virtual-switch | vpls | vrf);
      no-vrf-advertise;
      route-distinguisher (as-number:number | ip-address:number);
      vrf-import [ policy-names ];
      vrf-export [ policy-names ];
      vrf-table-label;
      vrf-target {
         export community-name;
         import community-name;
      }
      protocols {
         bgp {
            bgp-configuration;
         }
         isis {
            isis-configuration;
         }
         l2vpn {
            l2vpn-configuration;
         }
         ldp {
            ldp-configuration;
         }
         msdp {
            msdp-configuration;
         }
         ospf {
            domain-id domain-id;
            domain-vpn-tag number;
            route-type-community (vendor | iana);
```

```
        ospf-configuration;
    }
    ospf 3 {
        domain-id domain-id;
        domain-vpn-tag number;
        route-type-community (vendor | iana);
        ospf3-configuration;
    }
    pim {
        pim-configuration;
    }
    rip {
        rip-configuration;
    }
    vpls {
        vpls-configuration;
    }
}
routing-options {
    aggregate {
        defaults {
            ... aggregate-options ...
        }
        route destination-prefix {
            policy policy-name;
            ... aggregate-options ...
        }
    }
    auto-export {
        (disable | enable);
        family {
            inet {
                flow {
                    (disable | enable);
                    rib-group rib-group;
                }
                multicast {
                    (disable | enable);
                    rib-group rib-group;
                }
                unicast {
                    (disable | enable);
                    rib-group rib-group;
                }
            }
        }
        traceoptions {
            file filename <files number> <size size> <world-readable |
                no-world-readable>;
            flag flag <flag-modifier> <disable>;
        }
    }
    autonomous-system autonomous-system <loops number> {
        independent-domain;
    }
    confederation confederation-autonomous-systems
```

```
members autonomous-system;
dynamic-tunnels tunnel-name {
    destination-prefix prefix;
    source-address address;
    tunnel-type type-of-tunnel;
}
fate-sharing {
    group group-name;
    cost value;
    from address {
        to address;
    }
    flow {
    route name {
        match {
            match-conditions;
        }
        then {
            actions;
        }
    }
    validation {
        traceoptions {
            file filename <files number> <size size> <world-readable |
                no-world-readable>;
            flag flag <flag-modifier> <disable>;
        }
    }
    generate {
        defaults {
            generate-options;
        }
        route destination-prefix {
            policy policy-name;
            generate-options;
        }
    }
    instance-export [ policy-names ];
    instance-import [ policy-names ];
    interface-routes {
        family (inet | inet6) {
            export {
                lan;
                point-to-point;
            }
        }
        rib-group group-name;
    }
    martians {
        destination-prefix match-type <allow>;
    }
    maximum-paths path-limit <log-only | threshold value log-interval seconds>;
    maximum-prefixes prefix-limit <log-only | threshold value log-interval seconds>;
    multicast {
        forwarding-cache {
            threshold (suppress | reuse) value value;
```

```
    }
    interface interface-name {
        enable;
    }
    scope scope-name {
        interface interface-name;
        prefix destination-prefix;
    }
    scope-policy policy-name;
    ssm-groups {
        addresses;
    }
}
options {
    syslog (level level | upto level);
}
resolution {
    rib routing-table-name {
        import [ policy-names ];
        resolution-ribs [ routing-table-names ];
    }
}
rib routing-table-name {
    aggregate {
        defaults {
            ... aggregate-options ...
        }
        route destination-prefix {
            policy policy-name;
            ... aggregate-options ...
        }
    }
    filter {
        input filter-name;
    }
    generate {
        defaults {
            generate-options;
        }
        route destination-prefix {
            policy policy-name;
            generate-options;
        }
    }
    martians {
        destination-prefix match-type <allow>;
    }
    static {
        defaults {
            static-options;
        }
        passive group-name;
        route destination-prefix {
            lsp-next-hop {
                metric metric;
                preference preference;
```

```
            }
            next-hop;
            p2mp-lsp-next-hop {
               metric metric;
               preference preference;
            }
            qualified-next-hop address {
               interface interface-name;
               metric metric;
               preference preference;
            }
            static-options;
         }
      }
   }
   passive {
      group-name {
         import-policy [ policy-names ];
         import-rib [ group-names ];
         export-rib group-name;
      }
   }
   route-distinguisher-id address;
   route-record;
   router-id address;
   static {
      defaults {
         static-options;
      }
      rib-group group-name;
      route destination-prefix {
         bfd-liveness-detection {
            authentication {
               algorithm algorithm-name;
               key-chain key-chain-name;
               loose-check;
            }
            detection-time {
               threshold milliseconds;
            }
            local-address ip-address;
            minimum-interval milliseconds;
            minimum-receive-interval milliseconds;
            minimum-receive-ttl milliseconds;
            transmit-interval {
               threshold milliseconds;
               minimum-interval milliseconds;
            }
            multiplier number;
            version (1 | automatic);
         }
         lsp-next-hop {
            metric metric;
            preference preference;
         }
         next-hop;
```

```
                qualified-next-hop address {
                    interface interface-name;
                    metric metric;
                    preference preference;
                }
                static-options;
            }
        }
        traceoptions {
            file filename <files number> <size size> <world-readable |
                no-world-readable>;
            flag flag <flag-modifier> <disable>;
        }
    }
  }
}
```

The following statements can also be included at the [edit logical-systems *logical-system-name*] hierarchy level.

NOTE: The virtual-switch instance type is not supported at the [edit logical-systems *logical-system-name*] hierarchy level. For more detailed information about configuring a virtual switch on MX Series routers, see the *JUNOS MX Series Ethernet Services Routers Layer 2 Configuration Guide*.

[edit routing-options] Hierarchy Level

The following statements can also be included at the [edit logical-systems *logical-system-name*] hierarchy level.

```
routing-options {
    aggregate {
        defaults {
            ... aggregate-options ...
        }
        route destination-prefix {
            policy policy-name;
            ... aggregate-options ...
        }
    }
    auto-export {
        (disable | enable);
        family {
            inet {
                flow {
                    (disable | enable);
                    rib-group rib-group;
                }
                multicast {
                    (disable | enable);
                    rib-group rib-group;
```

```
        }
        unicast {
          (disable | enable);
          rib-group rib-group;
        }
      }
    }
    traceoptions {
      file filename <files number> <size size> <world-readable | no-world-readable>;
      flag flag <flag-modifier> <disable>;
    }
}
autonomous-system autonomous-system <loops number>;
confederation confederation-autonomous-system members autonomous-system;
dynamic-tunnels tunnel-name {
    destination-prefix prefix;
    source-address address;
    tunnel-type tunnel-type;
}
fate-sharing {
    group group-name;
    cost value;
    from address {
      to address;
    }
}
flow {
    route name {
      match {
        match-conditions;
      }
      then {
        actions;
      }
    }
    validation {
      traceoptions {
        file filename <files number> <size size> <world-readable | no-world-readable>;
        flag flag <flag-modifier> <disable>;
      }
    }
    forwarding-table {
      export [ policy-names ];
      (indirect-next-hop | no-indirect-next-hop);
      unicast-reverse-path (active-paths | feasible-paths);
    }
    generate {
      defaults {
        generate-options;
      }
      route destination-prefix {
        policy policy-name;
        generate-options;
      }
    }
    graceful-restart {
```

```
    disable;
    restart-duration seconds;
}
instance-export [ policy-names ];
instance-import [ policy-names ];
interface-routes {
    family (inet | inet6) {
        export {
            lan;
            point-to-point;
        }
    }
    rib-group group-name;
}
martians {
    destination-prefix match-type <allow>;
}
maximum-paths path-limit <log-only | threshold value log-interval seconds>;
maximum-prefixes prefix-limit <log-only | threshold value log-interval seconds>;
multicast {
    forwarding-cache {
        threshold (suppress | reuse) value value;
    }
    interface interface-name {
        enable;
    }
    scope scope-name {
        interface interface-name;
        prefix destination-prefix;
    }
    scope-policy policy-name;
    ssm-groups {
        address;
    }
}
options {
    syslog (level level | upto level);
}
ppm {
    delegate-processing;
}
resolution {
    rib routing-table-name {
        import [ policy-names ];
        resolution-ribs [ routing-table-names ];
    }
}
rib routing-table-name {
    aggregate {
        defaults {
            ... aggregate-options ...
        }
        rib-group group-name;
        route destination-prefix {
            policy policy-name;
            ... aggregate-options ...
```

```
        }
    }
    filter {
        input filter-name;
    }
    generate {
        defaults {
            generate-options;
        }
        route destination-prefix {
            policy policy-name;
            generate-options;
        }
    }
    martians {
        destination-prefix match-type <allow>;
    }
    static {
        defaults {
            static-options;
        }
        rib-group group-name;
        route destination-prefix {
            lsp-next-hop {
                metric metric;
                preference preference;
            }
            next-hop;
            qualified-next-hop address {
                interface interface-name;
                metric metric;
                preference preference;
            }
            static-options;
        }
    }
}
rib-groups {
    group-name {
        import-policy [ policy-names ];
        import-rib [ group-names ];
        export-rib group-name;
    }
}
route-distinguisher-id address;
route-record;
router-id address;
static {
    defaults {
        static-options;
    }
    rib-group group-name;
    route destination-prefix {
        bfd-liveness-detection {
            authentication {
                algorithm algorithm-name;
```

```
                key-chain key-chain-name;
                loose-check;
            }
            detection-time {
                threshold milliseconds;
            }
            holddown-interval milliseconds;
            local-address ip-address;
            minimum-interval milliseconds;
            minimum-receive-interval milliseconds;
            minimum-receive-ttl number;
            multiplier number;
            neighbor address;
            no-adaptation;
            transmit-interval {
                threshold milliseconds;
                minimum-interval milliseconds;
            }
            version (1 | automatic);
        }
        lsp-next-hop {
            metric metric;
            preference preference;
        }
        next-hop;
        p2mp-lsp-next-hop {
            metric metric;
            preference preference;
        }
        qualified-next-hop (address | interface-name) {
            interface interface-name;
            metric metric;
            preference preference;
        }
        source-routing {
            (ip | ipv6);
        }
        static-options;
    }
}
topologies {
    (inet | inet6) {
        topology name;
    }
}
traceoptions {
    file filename <files number> <size size> <world-readable | no-world-readable>;
    flag flag <flag-modifier> <disable>;
}
}
}
```

Part 2
Protocol-Independent Routing Properties

- Protocol-Independent Routing Properties Overview on page 45
- Configuring Routing Tables and Routes on page 51
- Configuring Other Protocol-Independent Routing Properties on page 111
- Configuring Logical Systems on page 135
- Summary of Protocol-Independent Routing Properties Configuration Statements on page 141

Chapter 3

Protocol-Independent Routing Properties Overview

This chapter discusses the following topics related to understanding and configuring protocol-independent routing properties:

- Protocol-Independent Routing Properties Configuration Statements on page 45
- Minimum Protocol-Independent Routing Properties Configuration on page 49

Protocol-Independent Routing Properties Configuration Statements

To configure protocol-independent routing properties, you include the following statements at the [edit routing-options] hierarchy level:

```
routing-options {
    aggregate {
        defaults {
            ... aggregate-options ...
        }
        route destination-prefix {
            policy policy-name;
            ... aggregate-options ...
        }
    }
    auto-export {
        (disable | enable);
        family {
            inet {
                multicast {
                    (disable | enable);
                    rib-group rib-group;
                }
                unicast {
                    (disable | enable);
                    rib-group rib-group;
                }
            }
        }
        traceoptions {
            file filename <files number> <size size> <world-readable | no-world-readable>;
            flag flag <flag-modifier> <disable>;
        }
```

```
}
autonomous-system autonomous-system <loops number>;
confederation confederation-autonomous-system members autonomous-system;
dynamic-tunnels tunnel-name {
   destination-prefix prefix;
   source-address address;
   tunnel-type type-of-tunnel;
}
fate-sharing {
   group group-name;
   cost value;
   from address {
      to address;
   }
}
forwarding-table {
   export [ policy-names ];
   (indirect-next-hop | no-indirect-next-hop);
   unicast-reverse-path (active-paths | feasible-paths);
}
generate {
   defaults {
      generate-options;
   }
   route destination-prefix {
      policy policy-name;
      generate-options;
   }
}
instance-export [ policy-names ];
instance-import [ policy-names ];
interface-routes {
   export {
      lan;
      point-to-point;
   }
   rib-group group-name;
}
martians {
   destination-prefix match-type <allow>;
}
maximum-paths route-limit <log-only | threshold value>;
multicast {
   forwarding-cache {
      threshold (suppress | reuse) value value;
   }
   interface interface-name;
   scope scope-name {
      interface [ interface-names ];
      prefix destination-prefix;
   }
   ssm-groups {
      address;
   }
}
ppm {
```

```
    no-delegate-processing;
}
resolution {
    rib routing-table-name {
        import [ policy-names ]
        resolution-ribs [ routing-table-names ];
    }
}
rib routing-table-name {
    aggregate {
        defaults {
            ... aggregate-options ...
        }
        route destination-prefix {
            policy policy-name;
            ... aggregate-options ...
        }
    }
    generate {
        defaults {
            generate-options;
        }
        route destination-prefix {
            policy policy-name;
            generate-options;
        }
    }
    martians {
        destination-prefix match-type <allow>;
    }
    source-routing {
        (ip | ipv6);
    }
    static {
        defaults {
            static-options;
        }
        rib-group group-name;
        route destination-prefix {
            bfd-liveness-detection {
                authentication {
                    algorithm algorithm-name;
                    key-chain key-chain-name;
                    loose-check;
                }
                detection-time {
                    threshold milliseconds;
                }
                <local-address ip-address>;
                minimum-interval milliseconds;
                minimum-receive-interval milliseconds;
                minimum-receive-ttl number;
                multiplier number;
                neighbor address;
                no-adaptation;
                transmit-interval {
```

```
                threshold milliseconds;
                minimum-interval milliseconds;
            }
            version (1 | automatic);
        }
        lsp-next-hop {
            metric metric;
            preference preference;
        }
        next-hop;
        p2mp-lsp-next-hop {
            metric metric;
            preference preference;
        }
        qualified-next-hop {
            interface interface-name;
            metric metric;
            preference preference;
        }
        static-options;
    }
  }
}
rib-groups {
  group-name {
    import-policy [ policy-names ];
    import-rib [ group-names ];
    export-rib group-name;
  }
}
route-record;
router-id address;
static {
  defaults {
    static-options;
  }
  rib-group group-name;
  route destination-prefix {
    bfd-liveness-detection {
      authentication {
        algorithm algorithm-name;
        key-chain key-chain-name;
        loose-check;
      }
      detection-time {
        threshold milliseconds;
      }
      <local-address ip-address>;
      minimum-interval milliseconds;
      minimum-receive-interval milliseconds;
      minimum-receive-ttl number;
      multiplier number;
      neighbor address;
      no-adaptation;
      transmit-interval {
        threshold milliseconds;
```

```
            minimum-interval milliseconds;
          }
          version (1 | automatic);
        }
        lsp-next-hop {
          metric metric;
          preference preference;
        }
        next-hop;
        qualified-next-hop {
          interface interface-name;
          metric metric;
          preference preference;
        }
        static-options;
      }
    }
    traceoptions {
      file filename <files number> <size size> <world-readable | no-world-readable>;
      flag flag <flag-modifier> <disable>;
    }
}
```

Minimum Protocol-Independent Routing Properties Configuration

All statements that configure protocol-independent routing properties are optional and do not have to be included in the configuration for the router to operate. However, if you are configuring BGP, you must configure an AS number and a router identifier. For OSPF, the router uses the IP address configured on the loopback interface (lo0) as the router identifier. If no IP address is configured on the loopback interface, the router uses the highest IP address for the router identifier.

Chapter 4

Configuring Routing Tables and Routes

This chapter discusses how to perform the following tasks for configuring routing tables and routes:

- Creating Routing Tables on page 52
- Configuring Static Routes on page 54
- Configuring the Destination of Static Routes on page 57
- Configuring the Next Hop for Static Routes on page 57
- Configuring an Independent Preference for Static Routes on page 58
- Specifying an LSP as the Next Hop for Static Routes on page 62
- Installing Static Routes into More than One Routing Table on page 63
- Configuring CLNS Static Routes on page 64
- Configuring Static Route Options on page 65
- Configuring Bidirectional Forwarding Detection on page 74
- Tracing BFD Protocol Traffic on page 78
- Overview of BFD Authentication for Static Routes on page 79
- Configuring BFD Authentication for Static Routes on page 81
- Configuring Default Routes on page 84
- Propagating Static Routes into Routing Protocols on page 85
- Examples: Configuring Static Routes on page 85
- Configuring Aggregate Routes on page 87
- Configuring the Destination of Aggregate Routes on page 89
- Configuring Aggregate Route Options on page 89
- Applying Policies to Aggregate Routes on page 94
- Advertising Aggregate Routes on page 95
- Configuring Generated Routes on page 96
- Configuring the Destination of Generated Routes on page 97
- Configuring Generated Route Options on page 97
- Applying Policies to Generated Routes on page 102
- Configuring Martian Addresses on page 103

- Configuring Flow Routes on page 105
- Applying Filters to the Forwarding Table on page 109

Creating Routing Tables

The JUNOS Software can maintain one or more routing tables, thus allowing the software to store route information learned from different protocols separately. For example, it is common for the routing software to maintain unicast routes and multicast routes in different routing tables. You also might have policy considerations that would lead you to create separate routing tables to manage the propagation of routing information.

Creating routing tables is optional. If you do not create any, the JUNOS Software uses its default routing tables, which are inet.0 for IP version 4 (IPv4) unicast routes, inet6.0 for IP version 6 (IPv6) unicast routes, inet.1 for the IPv4 multicast forwarding cache, and inet.3 for IPv4 MPLS. If Multiprotocol BGP (MBGP) is enabled, inet.2 is used for Subsequent Address Family Indicator (SAFI) 2 routes. If you configure a routing instance, the JUNOS Software creates the default unicast routing table instance-name.inet.0. If you configure a flow route, the JUNOS Software creates the flow routing table *instance-name*.inetflow.0.

If you want to add static, aggregate, generated, or martian routes only to the default IPv4 unicast routing table (inet.0), you do not have to create any routing tables because, by default, these routes are added to inet.0. You can add these routes just by including the static, aggregate, generate, and martians statements. For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

To explicitly create a routing table, include the rib statement:

```
rib routing-table-name {
    static {
        defaults {
            static-options;
        }
        rib-group group-name;
        route destination-prefix {
            lsp-next-hop lsp-name {
                metric metric;
                preference preference;
            }
            next-hop;
            p2mp-lsp-next-hop {
                metric metric;
                preference preference;
            }
            qualified-next-hop address {
                metric metric;
                preference preference;
            }
            static-options;
        }
    }
```

```
    aggregate {
        defaults {
            ... aggregate-options ...
        }
        route destination-prefix {
            policy policy-name;
            ... aggregate-options ...
        }
    }
    generate {
        defaults {
            generate-options;
        }
        route destination-prefix {
            policy policy-name;
            generate-options;
        }
    }
    martians {
        destination-prefix match-type <allow>;
    }
}
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

The routing table name, *routing-table-name*, includes the protocol family, optionally followed by a period and a number. The protocol family can be inet for the IPv4 family, inet6 for the IPv6 family, or iso for the International Standards Organization (ISO) protocol family. The number represents the routing instance. The first instance is 0.

Example: Creating Routing Tables

Create the IPv4 routing table inet.0 and add a static route to it:

```
[edit]
routing-options {
    rib inet.0 {
        static {
            route 140.122.0.0/16 next-hop 192.168.0.10;
        }
    }
}
```

Configure the primary IPv6 routing table inet6.0 and add a static route to it:

```
[edit routing-options]
rib inet6.0 {
    static {
        route 8:1::1/128 next-hop 8:3::1;
    }
}
```

Configuring Static Routes

The routing device uses dynamic routes to learn how to reach network destinations. Dynamic routes are determined from the information exchanged by the routing protocols and, as the name implies, the routes might change as network conditions change and these changes are discovered by the routing protocols. You can configure static (nonchanging) routes to some network destinations. The routing device uses static routes when it does not have a route to a destination that has a better (lower) preference value, when it cannot determine the route to a destination, or when it is forwarding unroutable packets.

Static routes are used when the network connects to a routing device or other system outside the network and either that system cannot run a routing protocol or you do not want to run a routing protocol on it. In these situations, a static route is created from an edge routing device to the outside system and then the edge routing device redistributes the static route to IGP.

A static route is installed in the routing table only when the route is active; that is, the list of next-hop routing devices configured for that route contains at least one next hop on an operational interface.

You can add the same routes to more than one routing table.

To configure static routes in the default IPv4 routing table (inet.0), include the static statement:

```
static {
    defaults {
        static-options;
    }
    rib-group group-name;
    route destination-prefix {
        bfd-liveness-detection {
            authentication {
                algorithm algorithm-name;
                key-chain key-chain-name;
                loose-check;
            }
            detection-time {
                threshold milliseconds;
            }
            <local-address ip-address>;
            minimum-interval milliseconds;
            minimum-receive-interval milliseconds;
            multiplier number;
            neighbor address;
            minimum-receive-ttl number;
            no-adaptation;
            transmit-interval {
                threshold milliseconds;
                minimum-interval milliseconds;
            }
            version (1 | automatic);
```

```
        }
        lsp-next-hop lsp-name {
            metric metric;
            preference preference;
        }
        next-hop address;
        next-hop options;
        p2mp-lsp-next-hop {
            metric metric;
            preference preference;
        }
        qualified-next-hop address {
            metric metric;
            preference preference;
        }
        static-options;
    }
}
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

To configure static routes in one of the other routing tables, to explicitly configure static routes in the default IPv4 route table (inet.0), or to explicitly configure static routes in the primary IPv6 routing table (inet6.0), include the static statement:

```
rib routing-table-name {
    static {
        defaults {
            static-options;
        }
        rib-group group-name;
        route destination-prefix {
            bfd-liveness-detection {
                authentication {
                    algorithm algorithm-name;
                    key-chain key-chain-name;
                    loose-check;
                }
                detection-time {
                    threshold milliseconds;
                }
                <local-address ip-address>;
                minimum-interval milliseconds;
                minimum-receive-interval milliseconds;
                minimum-receive-ttl number;
                multiplier number;
                neighbor address;
                no-adaptation;
                transmit-interval {
                    threshold milliseconds;
                    minimum-interval milliseconds;
                }
                version (1 | automatic);
            }
```

```
            lsp-next-hop lsp-name {
                metric metric;
                preference preference;
            }
            next-hop address;
            next-hop options;
            p2mp-lsp-next-hop {
                metric metric;
                preference preference;
            }
            qualified-next-hop address {
                metric metric;
                preference preference;
            }
            static-options;
        }
    }
}
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

NOTE: You cannot configure static routes for the IPv4 multicast routing table (inet.1) or the IPv6 multicast routing table (inet6.1).

The static statement consists of two parts:

- defaults—(Optional) Specify global static route options. These options only set default attributes inherited by all newly created static routes. These are treated as global defaults and apply to all the static routes you configure in the static statement.

NOTE: Specifying the global static route options does not create default routes. These options only set default attributes inherited by all newly created static routes.

- route—Configure individual static routes. In this part of the static statement, you optionally can configure static route options. These options apply to the individual destination only and override any options you configured in the defaults part of the static statement.

The following topics provide more information about configuring static routes:

- Configuring the Destination of Static Routes on page 57
- Configuring the Next Hop for Static Routes on page 57
- Configuring an Independent Preference for Static Routes on page 58
- Specifying an LSP as the Next Hop for Static Routes on page 62
- Installing Static Routes into More than One Routing Table on page 63
- Configuring CLNS Static Routes on page 64

- Configuring Static Route Options on page 65
- Configuring Default Routes on page 84
- Propagating Static Routes into Routing Protocols on page 85
- Examples: Configuring Static Routes on page 85

Configuring the Destination of Static Routes

When you configure an individual static route in the route part of the static statement, specify the destination of the route (in route *destination-prefix*) in one of the following ways:

- *network/mask-length*, where *network* is the network portion of the IP address and *mask-length* is the destination prefix length.
- default if this is the default route to the destination. This is equivalent to specifying an IP address of 0.0.0.0/0.

NOTE: IPv4 packets with a destination of 0.0.0.0 (the obsoleted limited broadcast address) and IPv6 packets with a destination of 0::0 are discarded by default. To forward traffic destined to these addresses, you can add a static route to 0.0.0.0/32 for IPv4 or 0::0/128 for IPv6.

Configuring the Next Hop for Static Routes

When you configure an individual static route in the route part of the static statement, specify how to reach the destination (in *next-hop*) in one of the following ways:

- next-hop *address*—IPv4 or IPv6 address of the next hop to the destination, specified as:
  - IPv4 or IPv6 address of the next hop
  - Interface name (for point-to-point interfaces only)
  - *address* or *interface-name* to specify an IP address of a multipoint interface or an interface name of a point-to-point interface.

NOTE: If an interface becomes unavailable, all configured static routes on that interface are withdrawn from the routing table.

- next-table *routing-table-name*—Name of the next routing table to the destination.

NOTE: Within a routing instance, you cannot configure a static route with the next-table inet.0 statement if any static route in the main routing instance is already configured with the next-table statement to point to the inet.0 routing table of the routing instance. For example, if you configure on the main routing instance a static route 192.168.88.88/32 with the next-table test.inet.0 statement and the routing instance test is also configured with a static route 192.168.88.88/32 with the next-table inet.0 statement, the commit operation fails. Instead, you must configure a routing table group both on the main instance and on the routing instance, which enables you to install the static route into both routing tables. For more information, see "Installing Static Routes into More than One Routing Table" on page 63.

- reject—Do not forward packets addressed to this destination. Instead, drop the packets, send ICMP (or ICMPv6) unreachable messages to the packets' originators, and install a reject route for this destination into the routing table.
- discard—Do not forward packets addressed to this destination. Instead, drop the packets, do not send ICMP (or ICMPv6) unreachable messages to the packets' originators, and install a reject route for this destination into the routing table.
- receive—Cause packets to the destination to be received by the local routing device.

Configuring an Independent Preference for Static Routes

Configuring independent preferences allows you to configure multiple static routes with different preferences and metrics to the same destination. The static route with the best preference, metric, and reachable next hop is chosen as the active route. This feature allows you to specify preference and metric on a next-hop basis using the qualified-next-hop statement.

NOTE: The preference and metric options configured by means of this statement only apply to the qualified next hops. The qualified next hop preference and metric override the route preference and metric (for that specific qualified next hop), similar to how the route preference overrides the default preference and metric (for that specific route).

To specify an independent preference for a static route on a point-to-point interface or on an Ethernet interface, include the following statements:

```
qualified-next-hop address {
    interface interface-name;
    metric metric;
    preference preference;
}
```

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for these statements.

Specify a next-hop interface by including the qualified-next-hop option. Specifying a next-hop interface is useful when you are creating a route to an IPv6 link-local next-hop address (which is a link-only scope address and is specific only to an interface). The preference value can be a number from 0 through 4,294,967,295 ($2^{32} - 1$). A lower number indicates a more preferred route. The metric value can also be a number from 0 through 4,294,967,295.

You can configure static routes on an unnumbered Ethernet interface by using the qualified-next-hop option to specify the unnumbered interface as the next-hop interface for a configured static route.

To configure an unnumbered Ethernet interface as the next-hop interface for a static route and to specify independent preferences, include the following statements:

```
qualified-next-hop interface-name {
    metric metric;
    preference preference;
}
```

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for these statements.

Keep the following points in mind when you configure static routes for unnumbered Ethernet interfaces:

- The prefix length of the static route must be 32.
- The routing device uses the Address Resolution Protocol (ARP) to resolve the media access control (MAC) address of the destination interface.

For information about how to configure an unnumbered Ethernet interface, see the *JUNOS Network Interfaces Configuration Guide*.

NOTE: The qualified-next-hop statement is mutually exclusive with all other types of next hops, except for next-hop address. Therefore, you cannot configure next-hop reject, next-hop discard, and next-hop receive with qualified-next-hop for the same destination.

For sample configurations, see the following sections:

- Example: Configuring Independent Preferences for an IPv4 Static Route on page 60
- Example: Configuring Independent Preferences for an IPv6 Static Route on page 60
- Example: Configuring Independent Preferences for an Unnumbered Ethernet Interface on page 61

Example: Configuring Independent Preferences for an IPv4 Static Route

The following example configures:

- A static route to 0.0.0.0/8 with a next hop through 192.168.1.254, with a metric of 10 and preference of 10.
- A static route to 10.0.0.0/8 with a next hop through 192.168.1.254, with a metric of 6 and preference of 5.
- A static route to 10.0.0.0/8 with a next hop through 192.168.1.2, with a metric of 6 and preference of 7.

```
[edit]
routing-options {
   static {
      defaults {
         metric 10;
         preference 10;
      }
      route 0.0.0.0/8 {
         next-hop 192.168.1.254 {
            retain;
            no-readvertise;
         }
         route 10.0.0.0/8 {
            next-hop [192.168.1.2];
            qualified-next-hop 192.168.1.254 {
               preference 5;
            }
            metric 6;
            preference 7;
         }
      }
   }
}
```

Example: Configuring Independent Preferences for an IPv6 Static Route

Configure the following qualified next hops:

- A static route to fec0:1:1:4::/64 with a next hop through fec0:1:1:2::1, with a metric 10 and preference 10.
- A static route to fec0:1:1:5::/64 with a next hop through fec0:1:1:2::2, with a metric 6 and preference 5.
- A static route to fec0:1:1:5::/64 with a next hop through fec0:1:1:2::3, with a metric 6 and preference 7.

```
[edit]
routing-options {
   rib inet6.0 {
      static {
```

```
        defaults {
            metric 10;
            preference 10;
        }
        route fec0:1:1:4::/64 {
            next-hop fec0:1:1:2::1 {
                retain;
                no-readvertise;
            }
            route fec0:1:1:5::/64 {
                next-hop fec0:1:1:2::3;
                qualified-next-hop fec0:1:1:2::2 {
                    preference 5;
                }
                metric 6;
                preference 7;
            }
        }
    }
  }
}
```

Example: Configuring Independent Preferences for an Unnumbered Ethernet Interface

The following example configures two things:

- An unnumbered Ethernet interface ge-0/0/0, which borrows an IP address from donor interface lo0.
- A static route to 7.7.7.1/32 with a next hop through unnumbered interface ge-0/0/0.0 with a with a metric of 5 and preference of 6.

```
interfaces {
    lo0 {
        unit 0 {
            family inet {
                address 5.5.5.1/32;
                address 6.6.6.1/32;
            }
        }
    }
}
interfaces
ge-0/0/0 {
    unit 0 {
        family inet {
            unnumbered-address lo0.0;
        }
    }
}
routing-options {
    static {
        route 7.7.7.1/32 {
            qualified next-hop ge-0/0/0.0 {
                metric 5;
```

```
            preference 6;
        }
    }
  }
}
```

Specifying an LSP as the Next Hop for Static Routes

Static routes can be configured with a next hop that is a label-switched path (LSP). This is useful when implementing filter-based forwarding. You can specify an LSP as the next hop and assign an independent preference and metric to this next hop.

To specify an LSP as the next hop for a static route, include the following statements:

```
lsp-next-hop lsp-name {
  metric metric;
  preference preference;
}
```

NOTE: The preference and metric configured by means of the lsp-next-hop statement only apply to the LSP next hops. The LSP next-hop preference and metric override the route preference and metric (for that specific LSP next hop), similar to how the route preference overrides the default preference and metric (for that specific route).

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for these statements.

The preference value can be a number in the range from 0 through 4,294,967,295 ($2^{32} - 1$) with a lower number indicating a more preferred route. The metric value can also be a number in the range from 0 through 4,294,967,295.

NOTE: The lsp-next-hop statement is mutually exclusive with all other types of next hops, except for next-hop address and qualified-next-hop. Therefore, you cannot configure next-hop reject, next-hop discard, next-hop receive, and next-table with lsp-next-hop for the same destination.

To specify a point-to-multipoint LSP as the next hop for a static route, include the following statements:

```
p2mp-lsp-next-hop {
  interface interface-name;
  metric metric;
  preference preference;
}
```

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for these statements.

Enable the qualified next-hop address on the interface by specifying the **interface** option. The **preference** value can be a number from 0 through 4,294,967,295 ($2^{32} - 1$). A lower number indicates a more preferred route. The **metric** value can also be a number from 0 through 4,294,967,295.

Installing Static Routes into More than One Routing Table

You can install a static route into more than one routing table. For example, you might want a simple configuration that allows you to install a static route into the default routing table **inet.0**, as well as a second routing table **inet.2**. Instead of configuring the same static route for each routing table, you can use routing table groups to insert the route into multiple tables. To create a routing table group, include the **rib-group** statement.

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

To install the routing table into a configured routing table group, include the **import-rib** statement:

```
rib-group group-name {
    import-rib [ routing-table-names ];
}
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

The first routing table you list in the **import-rib** statement must be the one you configured in the **rib-group** statement.

Examples: Installing a Static Route into More than One Routing Table

Install an IPv4 static route into **inet.0** and **inet.2**:

```
[edit routing-options rib table1.inet.0 static]
rib-group groupA;
[edit routing-options rib-groups]
groupA {
    import-rib [table1.inet.0 inet.0 inet.2];
}
```

Install an IPv6 static route into the **inet6.0** and **inet6.2** routing tables:

```
[edit routing-options rib table1.inet6.0 static]
rib-group groupA;
[edit routing-options rib-groups]
groupA {
    import-rib [table1.inet6.0 inet6.0 inet6.2];
}
```

Configuring CLNS Static Routes

Connectionless Network Services (CLNS) is an ISO Layer 3 protocol that uses network service access point (NSAP) reachability information instead of IPv4 prefixes. You can configure a static route for CLNS networks.

NOTE: CLNS is supported for J Series Services Routers only.

To configure a CLNS static route, include the following statements:

```
rib (iso.0 | instance-name.iso.0)
static {
   route nsap-prefix {
      next-hop (interface-name | iso-net);
      qualified-next-hop (interface-name | iso-net) {
         metric metric;
         preference preference;
      }
   }
}
```

For a list of hierarchy levels at which you can include these statements, see the CLNS statement summary sections in the *JUNOS Software Interfaces and Routing Configuration Guide*.

Specify the iso.0 routing table option to configure a primary instance CLNS static route. Specify the instance-name.iso.0 routing table option to configure CLNS static route for a particular routing instance. Specify the route *nsap-prefix* statement to configure the destination for the CLNS static route. Specify the next-hop (*interface-name* | *iso-net*) statement to configure the next hop, specified as an ISO network entity title (NET) or interface name. Include the qualified-next-hop (*interface-name* | *iso-net*) statement to configure the qualified next hop, specified as an ISO network entity title or interface name.

Example: Configuring a Static CLNS Route

Configure a static CLNS route with an NSAP of 47.0005.80ff.f800.0000.ffff.ffff:

```
[edit]
routing-options {
   rib iso.0 {
      static {
         iso-route 47.0005.80ff.f800.0000.ffff.ffff next-hop
            47.0005.80ff.f800.0000.0108.0001.1921.6800.4212;
         iso-route 47.0005.80ff.f800.0000.0108.0001.1921.6800.4212 next-hop
            t1-0/2/2.0;
         iso-route 47.0005.80ff.f800.0000.eee {
            qualified-next-hop 47.0005.80ff.f800.0000.0108.0001.1921.6800.4002
               {
               preference 20;
```

```
            metric 10;
        }
    }
  }
 }
}
```

For information on CLNS, see "Configuring CLNS for IS-IS" on page 356 and the *JUNOS Software Interfaces and Routing Configuration Guide*.

Configuring Static Route Options

In the **defaults** and **route** parts of the **static** statement, you can specify *static-options*, which define additional information about static routes that is included with the route when it is installed in the routing table. All static options are optional. Static options that you specify in the **defaults** part of the **static** statement are treated as global defaults and apply to all the static routes you configure in the **static** statement. Static options that you specify in the **route** part of the **static** statement override any global static options and apply to that destination only.

To configure static route options for IPv4 static routes, include one or more options in the **defaults** or **route** part of the **static** statement.

```
routing-options {
   static {
      defaults {
         (active | passive);
         as-path <as-path> <origin (egp | igp | incomplete)> <atomic-aggregate>
            <aggregator as-number in-address>;
         community [ community-ids ];
         (install | no-install);
         metric metric <type type>;
         (preference | preference2 | color | color2) preference <type type>;
         (readvertise | no-readvertise);
         (retain | no-retain);
         tag string;
      }
      rib-group group-name;
      route destination-prefix {
         (active | passive);
         as-path <as-path> <origin (egp | igp | incomplete)> <atomic-aggregate>
            <aggregator as-number in-address>;
         bfd-liveness-detection {
            authentication {
               algorithm algorithm-name;
               key-chain key-chain-name;
               loose-check;
            }
            detection-time {
               threshold milliseconds;
            }
            local-address ip-address;
            minimum-interval milliseconds;
            minimum-receive-interval milliseconds;
```

```
            minimum-receive-ttl number;
            multiplier number;
            neighbor address;
            no-adaptation;
            transmit-interval {
               threshold milliseconds;
               minimum-interval milliseconds
            }
            version (1 | automatic);
         }
         community [ community-ids ];
         (install | no-install);
         metric metric <type type>;
         (preference | preference2 | color | color2) preference <type type>;
         (readvertise | no-readvertise);
         resolve;
         (retain | no-retain);
         tag string;
      }
   }
}
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

To configure static route options for IPv6 static routes, include one or more options in the defaults or route part of the static statement. Each of these options is explained in the sections that follow.

```
rib inet6.0 {
   static {
      defaults {
         (active | passive);
         as-path <as-path> <origin (egp | igp | incomplete)> <atomic-aggregate>
            <aggregator as-number in-address>;
         community [ community-ids ];
         (install | metric);
         metric metric <type type>;
         (preference | preference2 | color | color2) preference <type type>;
         (readvertise | no-readvertise);
         resolve;
         (retain | no-retain);
      }
      rib-group group-name;
      route destination-prefix {
         (active | passive);
         as-path <as-path> <origin (egp | igp | incomplete)> <atomic-aggregate>
            <aggregator as-number in-address>;
         bfd-liveness-detection {
            authentication {
               algorithm algorithm-name;
               key-chain key-chain-name;
               loose-check;
            }
            detection-time {
```

```
            threshold milliseconds;
        }
        local-address ip-address;
        minimum-interval milliseconds;
        minimum-receive-interval milliseconds;
        minimum-receive-ttl number;
        multiplier number;
        neighbor address;
        no-adaptation;
        transmit-interval {
            threshold milliseconds;
            minimum-interval milliseconds;
        }
        version (1 | automatic);
    }
    community [ community-ids ];
    (install | no-install);
    metric metric <type type>;
    (preference | preference2 | color | color2) preference <type type>;
    (readvertise | no-readvertise);
    resolve;
    (retain | no-retain);
  }
 }
}
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

The following sections explain how to configure static route options:

- Configuring a Metric Value for Static Routes on page 67
- Configuring a Preference Value for Static Routes on page 68
- Associating BGP Communities with Static Routes on page 68
- Associating AS Paths with Static Routes on page 69
- Configuring an OSPF Tag String for Static Routes on page 70
- Controlling Temporary Installation of Static Routes in the Forwarding Table on page 71
- Controlling Retention of Static Routes in the Forwarding Table on page 71
- Controlling Retention of Inactive Static Routes in the Routing and Forwarding Tables on page 72
- Controlling Readvertisement of Static Routes on page 73
- Controlling Resolution of Static Routes to Prefixes That Are Not Directly Connected on page 74

Configuring a Metric Value for Static Routes

To associate a metric value with an IPv4 route, include the **metric** statement:

```
static (defaults | route) {
    metric metric <type type>;
```

```
}
```

To associate a metric value with an IPv6 route, include the **metric** statement:

```
rib inet6.0 static (defaults | route) {
   metric metric <type type>;
}
```

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for these statements.

In the **type** option, you can specify the type of route. For OSPF, when routes are exported to OSPF, type 1 routes are advertised in type 1 externals, and routes of any other type are advertised in type 2 externals. Note that if a qualified-next-hop metric value is configured, this value overrides the route metric.

Configuring a Preference Value for Static Routes

By default, static routes have a preference value of 5. To modify the default preference value, specify a primary preference value (**preference**). You also can specify a secondary preference value (**preference2**) and colors, which are even finer-grained preference values (**color** and **color2**). To do this for IPv4 static routes, include one or more of the following statements:

```
static (defaults | route) {
   (preference | preference2 | color | color2) preference <type type>;
}
```

To do this for IPv6 static routes, include one or more of the following statements:

```
rib inet6.0 static (defaults | route) {
   (preference | preference2 | color | color2) preference <type type>;
}
```

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for these statements.

The preference value can be a number in the range from 0 through 4,294,967,295 ($2^{32} - 1$) with a lower number indicating a more preferred route. For more information about preference values, see "Route Preferences Overview" on page 6. Note that if a qualified-next-hop preference value is configured, this value overrides the route preference.

In the **type** option, you can specify the type of route.

Associating BGP Communities with Static Routes

By default, no BGP community information is associated with static routes. To associate community information with IPv4 routes, include the **community** statement:

```
static (defaults | route) {
   community [ community-ids ];
}
```

To associate community information with IPv6 routes, include the community statement:

```
rib inet6.0 static (defaults | route) {
    community [ community-ids ];
}
```

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for these statements.

community-ids is one or more community identifiers for either communities or extended communities.

The format for community identifiers is:

as-number:*community-value*

as-number is the autonomous system (AS) number and can be a value in the range from 1 through 65,534. *community-value* is the community identifier and can be a number in the range from 0 through 65,535.

You also can specify *community-ids* as one of the following well-known community names, which are defined in RFC 1997:

- no-export—Routes containing this community name are not advertised outside a BGP confederation boundary.
- no-advertise—Routes containing this community name are not advertised to other BGP peers.
- no-export-subconfed—Routes containing this community name are not advertised to external BGP peers, including peers in other members' ASs inside a BGP confederation.

You can also explicitly exclude BGP community information with a static route using the none option. Include none when configuring an individual route in the route portion of the static statement to override a community option specified in the defaults portion of the statement.

NOTE: Extended community attributes are not supported at the [edit routing-options] hierarchy level. You must configure extended communities at the [edit policy-options] hierarchy level. For information about configuring extended communities information, see the "Defining BGP Communities and Extended Communities for Use in Routing Policy Match Conditions" section in the *JUNOS Policy Framework Configuration Guide*.

Associating AS Paths with Static Routes

By default, no AS path information is associated with static routes. To associate AS path information with IPv4 routes, include the as-path statement:

```
static (defaults | route) {
```

```
    as-path <as-path> <origin (egp | igp | incomplete)> <atomic-aggregate> <aggregator
      as-number in-address>;
}
```

To associate AS path information with IPv6 routes, include the **as-path** statement:

```
rib inet6.0 static (defaults | route) {
    as-path <as-path> <origin (egp | igp | incomplete)> <atomic-aggregate> <aggregator
      as-number in-address>;
}
```

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for these statements.

as-path is the AS path to include with the route. It can include a combination of individual AS path numbers and AS sets. Enclose sets in brackets ([ ]). The first AS number in the path represents the AS immediately adjacent to the local AS. Each subsequent number represents an AS that is progressively farther from the local AS, heading toward the origin of the path.

In JUNOS Release 9.1 and later, the range that you can configure for the AS number has been extended to provide BGP support for 4-byte AS numbers as defined in RFC 4893, *BGP Support for Four-octet AS Number Space*. You can now configure a number from 1 through 4,294,967,295. All releases of the JUNOS Software support 2-byte AS numbers.

You also can specify the AS path using the BGP origin attribute, which indicates the origin of the AS path information:

- **igp**—Path information originated within the local AS.
- **egp**—Path information originated in another AS.
- **incomplete**—Path information learned by some other means.

To attach the BGP **ATOMIC_AGGREGATE** path attribute to the static route, specify the **atomic-aggregate** statement. This path attribute indicates that the local system selected a less specific route rather than a more specific route.

To attach the BGP **AGGREGATOR** path attribute to the static route, specify the **aggregator** statement. When using this statement, you must specify the last AS number that formed the static route (encoded as two octets), followed by the IP address of the BGP system that formed the static route.

Configuring an OSPF Tag String for Static Routes

By default, no OSPF tag strings are associated with static routes. You can specify an OSPF tag string by including the **tag** statement:

```
static (defaults | route) {
    tag string;
}
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

Controlling Temporary Installation of Static Routes in the Forwarding Table

By default, the JUNOS Software installs all active static routes into the forwarding table. To configure the software not to install active IPv4 static routes into the forwarding table, include the **no-install** statement:

```
static (defaults | route) {
    no-install;
}
```

To configure the software not to install active IPv6 static routes into the forwarding table, include the **no-install** statement:

```
rib inet6.0 static (defaults | route) {
    no-install;
}
```

Even if you configure a route so it is not installed in the forwarding table, the route is still eligible to be exported from the routing table to other protocols. To explicitly install IPv4 routes into the forwarding table, include the **install** statement. Include this statement when configuring an individual route in the **route** portion of the **static** statement to override a **no-install** option specified in the **defaults** portion of the statement.

```
static (defaults | route) {
    install;
}
```

To explicitly install IPv6 routes into the forwarding table, include the **install** statement. Include this statement when configuring an individual route in the **route** portion of the **static** statement to override a **no-install** statement specified in the **defaults** portion of the statement.

```
rib inet6.0 static (defaults | route) {
    install;
}
```

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for these statements.

Controlling Retention of Static Routes in the Forwarding Table

By default, statically configured routes are deleted from the forwarding table when the routing protocol process shuts down normally. To have an IPv4 static route remain in the forwarding table, include the **retain** statement. Doing this greatly reduces the time required to restart a system that has a large number of routes in its routing table.

```
static (defaults | route) {
    retain;
}
```

To have an IPv6 static route remain in the forwarding table, include the retain statement. Doing this greatly reduces the time required to restart a system that has a large number of routes in its routing table.

```
rib inet6.0 static (defaults | route) {
    retain;
}
```

To explicitly specify that IPv4 routes be deleted from the forwarding table, include the no-retain statement. Include this statement when configuring an individual route in the route portion of the static statement to override a retain option specified in the defaults portion of the statement.

```
static (defaults | route) {
    no-retain;
}
```

To explicitly specify that IPv6 routes be deleted from the forwarding table, include the no-retain statement. Include this statement when configuring an individual route in the route portion of the static statement to override a retain statement specified in the defaults portion of the statement.

```
rib inet6.0 static (defaults | route) {
    no-retain;
}
```

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for these statements.

Controlling Retention of Inactive Static Routes in the Routing and Forwarding Tables

Static routes are only removed from the routing table if the next hop becomes unreachable. This can occur if the local or neighbor interface goes down. To have an IPv4 static route remain installed in the routing and forwarding tables, include the passive statement:

```
static (defaults | route) {
    passive;
}
```

To have an IPv6 static route remain installed in the routing and forwarding tables, include the passive statement:

```
rib inet6.0 static (defaults | route) {
    passive;
}
```

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for these statements.

Routes that have been configured to remain continually installed in the routing and forwarding tables are marked with reject next hops when they are inactive.

To explicitly remove IPv4 static routes when they become inactive, include the active statement. Include this statement when configuring an individual route in the route portion of the static statement to override a passive option specified in the defaults portion of the statement.

```
static (defaults | route) {
   active;
}
```

To explicitly remove IPv6 static routes when they become inactive, include the active statement. Include this statement when configuring an individual route in the route portion of the static statement to override a passive statement specified in the defaults portion of the statement.

```
rib inet6.0 static (defaults | route) {
   active;
}
```

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for these statements.

Controlling Readvertisement of Static Routes

By default, static routes are eligible to be readvertised (that is, exported) by dynamic routing protocols. To mark an IPv4 static route as being ineligible for readvertisement, include the no-readvertise statement:

```
static (defaults | route) {
   no-readvertise;
}
```

To mark an IPv6 static route as being ineligible for readvertisement, include the no-readvertise statement:

```
rib inet6.0 static (defaults | route) {
   no-readvertise;
}
```

To explicitly readvertise IPv4 static routes, include the readvertise statement. Include the readvertise statement when configuring an individual route in the route portion of the static statement to override a no-readvertise statement specified in the defaults portion of the statement.

```
static (defaults | route) {
   readvertise;
}
```

To explicitly readvertise IPv6 static routes, include the readvertise statement. Include the readvertise statement when configuring an individual route in the route portion of the static statement to override a no-readvertise option specified in the defaults portion of the statement.

```
rib inet6.0 static (defaults | route) {
   readvertise;
```

```
}
```

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for these statements.

Controlling Resolution of Static Routes to Prefixes That Are Not Directly Connected

By default, static routes can point only to a directly connected next hop. You can configure an IPv4 route to a prefix that is not directly connected by resolving the route through the inet.0 and inet.3 routing tables. To configure an IPv4 static route to a prefix that is not a directly connected next hop, include the resolve statement:

```
static (defaults | route) {
   resolve;
}
```

You can configure an IPv6 route to a prefix that is not directly connected by resolving the route through the inet6.0 routing table. To configure an IPv6 static route to a prefix that is not a directly connected next hop, include the resolve statement:

```
rib inet6.0 static (defaults | route) {
   resolve;
}
```

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for these statements.

Configuring Bidirectional Forwarding Detection

The Bidirectional Forwarding Detection (BFD) protocol is a simple hello mechanism that detects failures in a network. Hello packets are sent at a specified, regular interval. A neighbor failure is detected when the routing device stops receiving a reply after a specified interval. BFD works with a wide variety of network environments and topologies. The failure detection timers for BFD have shorter time limits than the failure detection mechanisms of static routes, providing faster detection. These timers are also adaptive. For example, a timer can adapt to a higher value if an adjacency fails, or a neighbor can negotiate a higher value than the one configured. By default, BFD is supported on single-hop static routes. In JUNOS Release 8.2 and later, BFD also supports multihop static routes.

To enable failure detection, include the bfd-liveness-detection statement:

```
static route destination-prefix {
   bfd-liveness-detection {
      detection-time {
         threshold milliseconds;
      }
      local-address ip-address;
      minimum-interval milliseconds;
      minimum-receive-interval milliseconds;
      minimum-receive-ttl number;
      multiplier number;
      neighbor address;
```

```
        no-adaptation;
        transmit-interval {
            threshold milliseconds;
            minimum-interval milliseconds;
        }
        version (1 | automatic);
    }
}
```

In JUNOS Release 9.1 and later, the BFD protocol is supported for IPv6 static routes. Global unicast and link-local IPv6 addresses are supported for static routes. The BFD protocol is not supported on multicast or anycast IPv6 addresses. For IPv6, the BFD protocol supports only static routes and only in JUNOS Release 9.3 and later. IPv6 for BFD is not supported for any other protocol. To configure the BFD protocol for IPv6 static routes, include the bfd-liveness-detection statement at the [edit routing-options rib inet6.0 static route *destination-prefix*] hierarchy level.

In JUNOS Release 8.5 and later, you can configure a hold-down interval to specify how long the BFD session must remain up before state change notification is sent. To specify the hold-down interval, include the holddown-interval statement:

```
static route destination-prefix {
    bfd-liveness-detection {
        holddown-interval milliseconds;
    }
}
```

You can configure a number in the range from 0 through 255,000 milliseconds, and the default is 0. If the BFD session goes down and then comes back up during the hold-down interval, the timer is restarted.

NOTE: If a single BFD session includes multiple static routes, the hold-down interval with the highest value is used.

To specify the minimum transmit and receive intervals for failure detection, include the minimum-interval statement:

```
static route destination-prefix {
    bfd-liveness-detection {
        minimum-interval milliseconds;
    }
}
```

This value represents the minimum interval at which the local routing device transmits hello intervals as well as the minimum interval that the routing device expects to receive a reply from a neighbor with which it has established a BFD session. You can configure a number in the range from 1 through 255,000 milliseconds. You can also specify the minimum transmit and receive intervals separately.

To specify only the minimum receive interval for failure detection, include the minimum-receive-interval statement:

```
static route destination-prefix {
```

```
    bfd-liveness-detection {
        minimum-receive-interval milliseconds;
    }
}
```

This value represents the minimum interval at which the local routing device expects to receive a reply from a neighbor with which it has established a BFD session. You can configure a number in the range from 1 through 255,000 milliseconds.

To specify the number of hello packets not received by the neighbor that causes the originating interface to be declared down, include the multiplier statement:

```
static route destination-prefix {
    bfd-liveness-detection {
        multiplier number;
    }
}
```

The default value is 3. You can configure a number in the range from 1 through 255.

To specify a threshold for detecting the adaptation of the detection time, include the threshold statement:

```
static route destination-prefix {
    bfd-liveness-detection {
        detection-time {
            threshold milliseconds;
        }
    }
}
```

When the BFD session detection time adapts to a value equal to or higher than the threshold, a single trap and a system log message are sent. The detection time is based on the multiplier of the minimum-interval or the minimum-receive-interval value. The threshold must be a higher value than the multiplier for either of these configured values. For example if the minimum-receive-interval is 300 ms and the multiplier is 3, the total detection time is 900 ms. Therefore, the detection time threshold must have a value higher than 900.

To specify only the minimum transmit interval for failure detection, include the transmit-interval minimum-interval statement:

```
static route destination-prefix {
    bfd-liveness-detection {
        transmit-interval {
            minimum-interval milliseconds;
        }
    }
}
```

This value represents the minimum interval at which the local routing device transmits hello packets to the neighbor with which it has established a BFD session. You can configure a value in the range from 1 through 255,000 milliseconds.

To specify the transmit threshold for detecting the adaptation of the transmit interval, include the transmit-interval threshold statement:

```
static route destination-prefix {
    bfd-liveness-detection {
        transmit-interval {
            threshold milliseconds;
        }
    }
}
```

The threshold value must be greater than the transmit interval. When the BFD session detection time adapts to a value higher than the threshold, a single trap and a system log message are sent. The detection time is based on the multiplier of the minimum-interval or the minimum-receive-interval value. The threshold must be a higher value than the multiplier for either of these configured values.

To specify the BFD version, include the version statement:

```
static route destination-prefix {
    bfd-liveness-detection {
        version (1 | automatic);
    }
}
```

The default is to have the version detected automatically.

To include an IP address for the next hop of the BFD session, include the neighbor statement:

```
static route destination-prefix {
    next-hop interface-name;
    bfd-liveness-detection {
        neighbor address;
    }
}
```

NOTE: You must configure the neighbor statement if the next hop specified is an interface name. If you specify an IP address as the next hop, that address is used as the neighbor address for the BFD session.

In JUNOS Release 9.0 and later, you can configure BFD sessions not to adapt to changing network conditions. To disable BFD adaptation, include the no-adaptation statement:

```
bfd-liveness-detection {
    no-adaptation;
}
```

NOTE: We recommend that you not disable BFD adaptation unless it is preferable not to have BFD adaptation in your network.

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for these statements.

NOTE: If BFD is configured only on one end of a static route, the route is removed from the routing table. BFD establishes a session when BFD is configured on both ends of the static route.

BFD is not supported on ISO address families in static routes. BFD does support IS-IS.

If you configure graceful Routing Engine switchover at the same time as BFD, graceful Routing Engine switchover does not preserve the BFD state information during a failover.

The JUNOS Software also supports BFD over multihop static routes. For example, you can configure BFD over a Layer 3 path to provide path integrity over that path. You can limit the number of hops by specifying the time-to-live (TTL).

To configure BFD over multihop static routes, include the following statements:

```
static route destination-prefix {
    bfd-liveness-detection {
        local-address ip-address;
        minimum-receive-ttl number;
    }
}
```

To specify the source address for the multihop static route and to enable multihop BFD support, include the local-address statement.

To specify the number of hops, include the minimum-receive-ttl statement. You must configure this statement for a multihop BFD session. You can configure a value in the range from 1 through 255. It is optional for a single-hop BFD session. If you configure the minimum-receive-ttl statement for a single-hop session, the value must be 255.

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for these statements.

Related Topics

- Tracing BFD Protocol Traffic on page 78
- Overview of BFD Authentication for Static Routes on page 79
- Configuring BFD Authentication for Static Routes on page 81

Tracing BFD Protocol Traffic

To trace BFD protocol traffic, you can specify options in the global traceoptions statement at the [edit routing-options] hierarchy level, and you can specify BFD-specific options by including the traceoptions statement at the [edit protocols bfd hierarchy level.

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

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

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

- adjacency—Trace adjacency messages.
- all—Trace all options.
- error—Trace all error messages.
- event—Trace all events.
- issu—Trace in-service software upgrade (ISSU) packet activity.
- nsr-packet—Trace active nonstop active routing (NSR) packet activity.
- nsr-synchronization—Trace NSR synchronization events.
- packet—Trace all packets.
- pipe—Trace pipe messages.
- pipe-detail—Trace pipe messages in detail.
- ppm-packet—Trace packet activity by periodic packet management (PPM).
- state—Trace state transitions.
- timer—Trace timer processing.

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

For general information about tracing, see the tracing and logging information in the *JUNOS System Basics Configuration Guide*.

Related Topics

- Configuring Bidirectional Forwarding Detection on page 74
- Overview of BFD Authentication for Static Routes on page 79
- Configuring BFD Authentication for Static Routes on page 81

Overview of BFD Authentication for Static Routes

BFD enables rapid detection of communication failures between adjacent systems. By default, authentication for BFD sessions is disabled. However, when you run BFD over Network Layer protocols, the risk of service attacks can be significant. We strongly recommend using authentication if you are running BFD over multiple hops

or through insecure tunnels. Beginning with JUNOS Release 9.6, the JUNOS Software supports authentication for BFD sessions running over IPv4 and IPv6 static routes. BFD authentication is not supported on MPLS OAM sessions. BFD authentication is only supported in the domestic image and is not available in the export image.

You authenticate BFD sessions by specifying an authentication algorithm and keychain, and then associating that configuration information with a security authentication keychain using the keychain name.

The following sections describe the supported authentication algorithms, security keychains, and level of authentication that can be configured:

- BFD Authentication Algorithms on page 80
- Security Authentication Keychains on page 81
- Strict Versus Loose Authentication on page 81

BFD Authentication Algorithms

JUNOS Software supports the following algorithms for BFD authentication:

- simple-password—Plain-text password. One to 16 bytes of plain text are used to authenticate the BFD session. One or more passwords may be configured. This method is the least secure and should be used only when BFD sessions are not subject to packet interception.
- keyed-md5—Keyed Message Digest 5 hash algorithm for sessions with transmit and receive intervals greater than 100 ms. To authenticate the BFD session, keyed MD5 uses one or more secret keys (generated by the algorithm) and a sequence number that is updated periodically. With this method, packets are accepted at the receiving end of the session if one of the keys matches and the sequence number is greater than or equal to the last sequence number received. Although more secure than a simple password, this method is vulnerable to replay attacks. Increasing the rate at which the sequence number is updated can reduce this risk.
- meticulous-keyed-md5—Meticulous keyed Message Digest 5 hash algorithm. This method works in the same manner as keyed MD5, but the sequence number is updated with every packet. Although more secure than keyed MD5 and simple passwords, this method may take additional time to authenticate the session.
- keyed-sha-1—Keyed Secure Hash Algorithm I for sessions with transmit and receive intervals greater than 100 ms. To authenticate the BFD session, keyed SHA uses one or more secret keys (generated by the algorithm) and a sequence number that is updated periodically. The key is not carried within the packets. With this method, packets are accepted at the receiving end of the session if one of the keys matches and the sequence number is greater than the last sequence number received.
- meticulous-keyed-sha-1—Meticulous keyed Secure Hash Algorithm I. This method works in the same manner as keyed SHA, but the sequence number is updated with every packet. Although more secure than keyed SHA and simple passwords, this method may take additional time to authenticate the session.

NOTE: Nonstop active routing (NSR) is not supported with meticulous-keyed-md5 and meticulous-keyed-sha-1 authentication algorithms. BFD sessions using these algortihms may go down after a switchover.

Security Authentication Keychains

The security authentication keychain defines the authentication attributes used for authentication key updates. When the security authentication keychain is configured and associated with a protocol through the keychain name, authentication key updates can occur without interrupting routing and signaling protocols.

The authentication keychain contains one or more keychains. Each keychain contains one or more keys. Each key holds the secret data and the time at which the key becomes valid. The algorithm and keychain must be configured on both ends of the BFD session, and they must match. Any mismatch in configuration prevents the BFD session from being created.

BFD allows multiple clients per session, and each client can have its own keychain and algorithm defined. To avoid confusion, we recommend specifying only one security authentication keychain.

Strict Versus Loose Authentication

By default, strict authentication is enabled and authentication is checked at both ends of each BFD session. Optionally, to smooth migration from nonauthenticated sessions to authenticated sessions, you can configure *loose checking*. When loose checking is configured, packets are accepted without authentication being checked at each end of the session. This feature is intended for transitional periods only.

Related Topics

- Configuring BFD Authentication for Static Routes on page 81
- bfd-liveness-detection statement
- authentication-key-chains statement in the *JUNOS System Basics Configuration Guide*
- show bfd session command in the *JUNOS Routing Protocols and Policies Command Reference*
- Configuring Bidirectional Forwarding Detection on page 74

Configuring BFD Authentication for Static Routes

Beginning with JUNOS Release 9.6, you can configure authentication for BFD sessions running over IPv4 and IPv6 static routes. Routing instances are also supported. Only three steps are needed to configure authentication on a BFD session:

1. Specify the BFD authentication algorithm for the static route.
2. Associate the authentication keychain with the static route.
3. Configure the related security authentication keychain.

The following sections provide instructions for configuring and viewing BFD authentication on static routes:

- Configuring the BFD Authentication Parameters on page 82
- Viewing Authentication Information for BFD Sessions on page 83

Configuring the BFD Authentication Parameters

To configure BFD authentication:

1. Specify the algorithm (keyed-md5, keyed-sha-1, meticulous-keyed-md5, meticulous-keyed-sha-1, or simple-password) to use for BFD authentication on a static route or routing instance.

   ```
   [edit]
   user@host# set routing-options static route ipv4 bfd-liveness-detection
      authentication algorithm keyed-sha-1
   ```

NOTE: Nonstop active routing (NSR) is not supported with meticulous-keyed-md5 and meticulous-keyed-sha-1 authentication algorithms. BFD sessions using these algorithms may go down after a switchover.

2. Specify the keychain to be used to associate BFD sessions on the specified route or routing instance with the unique security authentication keychain attributes. This should match the keychain name configured at the [edit security authentication key-chains] hierarchy level.

   ```
   [edit]
   user@host# set routing-options static route ipv4 bfd-liveness-detection
      authentication keychain bfd-sr4
   ```

NOTE: The algorithm and keychain must be configured on both ends of the BFD session, and they must match. Any mismatch in configuration prevents the BFD session from being created.

3. Specify the unique security authentication information for BFD sessions:
   - The matching *key-chain-name* as specified in step 2.
   - At least one *key*, a unique integer between 0 and 63. Creating multiple keys allows multiple clients to use the BFD session.
   - The *secret-data* used to allow access to the session.
   - The time at which the authentication key becomes active, *yyyy-mm-dd.hh:mm:ss*.

   ```
   [edit security]
   user@host# authentication-key-chains key-chain bfd-sr4 key 53 secret
      $9$ggaJDmPQ6/tJgF/AtREVsyPsnCtUHm start-time 2009-06-14.10:00:00
   ```

4. (Optional) Specify loose authentication checking if you are transitioning from nonauthenticated sessions to authenticated sessions.

```
[edit]
user@host> set routing-options static route ipv4 bfd-liveness-detection
   authentication loose-check
```

5. (Optional) View your configuration using the show bfd session detail or show bfd session extensive command.

6. Repeat these steps to configure the other end of the BFD session.

NOTE: BFD authentication is only supported in the domestic image and is not available in the export image.

Viewing Authentication Information for BFD Sessions

You can view the existing BFD authentication configuration using the show bfd session detail and show bfd session extensive commands.

The following example shows BFD authentication configured for the static route at 192.168.208.26. It specifies the keyed SHA-1 authentication algorithm and a keychain name of bfd-static. The authentication keychain is configured with two keys. Key 1 contains the secret data "9ggaJDmPQ6/tJgF/AtREVsyPsnCtUHm" and a start time of June 1, 2009, at 9:46:02 AM PST. Key 2 contains the secret data "9a5jiKW9I.reP38ny.TszF2/9" and a start time of June 1, 2009, at 3:29:20 PM PST.

```
[edit routing-options]
static route 192.168.208.26 {
   bfd-liveness-detection {
      authentication {
         algorithm keyed-sha-1;
         key-chain bfd-static;
      }
   }
}
[edit security]
authentication key-chains {
   key-chain bfd-static {
      key 1 {
         secret "$9$ggaJDmPQ6/tJgF/AtREVsyPsnCtUHm";
         start-time "2009-6-1.09:46:02 -0700";
      }
      key 2 {
         secret "$9$a5jiKW9I.reP38ny.TszF2/9";
         start-time "2009-6-1.15:29:20 -0700";
      }
   }
}
```

If you commit these updates to your configuration, you would see output similar to the following. In the output for the show bfd sessions detail command, Authenticate

is displayed to indicate that BFD authentication is configured. For more information about the configuration, use the show bfd sessions extensive command. The output for this command provides the keychain name, the authentication algorithm and mode for each client in the session, and the overall BFD authentication configuration status, keychain name, and authentication algorithm and mode.

show bfd sessions detail

```
user@host# show bfd session detail

                                                  Detect   Transmit
Address                 State     Interface       Time     Interval  Multiplier
192.168.208.26          Up        so-1/0/0.0      2.400     0.800        10
 Client Static, TX interval 0.600, RX interval 0.600, Authenticate
 Session up time 00:18:07
 Local diagnostic None, remote diagnostic NbrSignal
 Remote state Up, version 1
 Replicated

1 sessions, 1 clients
Cumulative transmit rate 1.2 pps, cumulative receive rate 1.2 pps
```

show bfd sessions extensive

```
user@host# show bfd session extensive
                                                  Detect   Transmit
Address                 State     Interface       Time     Interval  Multiplier
192.168.208.26          Up        so-1/0/0.0      2.400     0.800        10
 Client Static, TX interval 0.600, RX interval 0.600, Authenticate
       keychain bfd-static, algo keyed-md5, mode loose
 Session up time 00:18:07
 Local diagnostic None, remote diagnostic NbrSignal
 Remote state Up, version 1
 Replicated
 Min async interval 0.600, min slow interval 1.000
 Adaptive async TX interval 0.600, RX interval 0.600
 Local min TX interval 0.600, minimum RX interval 0.600, multiplier 10
 Remote min TX interval 0.800, min RX interval 0.800, multiplier 3
 Local discriminator 2, remote discriminator 3
 Echo mode disabled/inactive
 Authentication enabled/active, keychain bfd-static, algo keyed-md5, mode loose

1 sessions, 1 clients
Cumulative transmit rate 1.2 pps, cumulative receive rate 1.2 pps
```

Related Topics

- Overview of BFD Authentication for Static Routes on page 79
- bfd-liveness-detection statement
- authentication-key-chains statement in the *JUNOS System Basics Configuration Guide*
- show bfd session command in the *JUNOS Routing Protocols and Policies Command Reference*
- Configuring Bidirectional Forwarding Detection on page 74

Configuring Default Routes

To configure an IPv4 default route, include the next-hop *address* and retain statements:

```
static route default {
```

```
    next-hop address;
    retain;
}
```

To configure an IPv6 static route, include the next-hop *address* and retain statements:

```
rib inet6.0 static (default | route) {
    next-hop address;
    retain;
}
```

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for these statements.

Propagating Static Routes into Routing Protocols

A common way to propagate static routes into the various routing protocols is to configure the routes so that the next-hop routing device is the loopback address (commonly, 127.0.0.1). However, configuring static routes in this way with the JUNOS Software (by including a statement such as route *address/mask-length* next-hop 127.0.0.1) does not propagate the static routes, because the forwarding table ignores static routes whose next-hop routing device is the loopback address. To propagate IPv4 static routes into the routing protocols, include the discard statement:

```
rib inet.0 static (defaults | route) {
    discard;
}
```

To propagate IPv6 static routes into the routing protocols, include the discard statement:

```
rib inet6.0 static (defaults | route) {
    discard;
}
```

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for these statements.

In this configuration, you use the discard option instead of reject because discard does not send an ICMP (or ICMPv6) unreachable message for each packet that it drops.

Examples: Configuring Static Routes

Configure an IPv4 default route through the next-hop router 192.238.52.33:

```
[edit]
user@host# set routing-options static route 0.0.0.0/0 next-hop 192.238.52.33
[edit]
user@host# show
routing-options {
    static {
```

```
        route 0.0.0.0/0 next-hop 192.238.52.33;
    }
}
```

Configure IPv4 static routes that are retained in the forwarding table when the routing software shuts down normally:

```
[edit]
user@host# set routing-options static route 0.0.0.0/0 next-hop 192.168.1.254
    retain
[edit]
user@host# set routing-options static route 10.1.1.1/32 next-hop 127.0.0.1 retain
[edit]
user@host# show
routing-options {
    static {
        route 0.0.0.0/0 {
            next-hop 192.168.1.254;
            retain;
        }
        route 10.1.1.1/32 {
            next-hop 127.0.0.1;
            retain;
        }
    }
}
```

Configure an IPv4 static route and have it propagate into the routing protocols. In this example, specify that the route 143.172.0.0/6 next-hop 127.0.0.1 should be discarded.

```
[edit]
user@host# set routing-options static route 143.172.0.0/6 discard
[edit]
user@host# show
routing-options {
    static {
        route 143.172.0.0/6 discard;
    }
}
```

Install an IPv4 static route into both inet.0 and inet.2:

```
[edit]
user@host# set routing-options static rib-group some-group
user@host# set rib-groups some-group import-rib [inet.0 inet.2]
[edit]
user@host# show
routing-options {
    static {
        rib-group some-group;
    }
    rib-groups {
        some-group {
            import-rib [ inet.0 inet.2 ];
        }
```

```
    }
}
```

Configure an IPv6 default route through the next-hop router 8:3::1:

```
[edit]
user@host# set routing-options rib inet6.0 static route 0::/0 next-hop 8:3::1
[edit]
user@host# show
routing-options {
    rib inet6.0 static {
        route abcd::/48 next-hop 8:3::1;
    }
}
```

Resolve an IPv6 static route to non-next-hop router 1::/64 using next-hop router 2000::1:

```
[edit]
user@host# set routing-options rib inet6.0 static route 1::/64 next-hop 2000::1
    resolve
[edit]
user@host# show route 1::/64
inet6.0: 26 destinations, 27 routes (25 active, 0 holddown, 1 hidden)
+ = Active Route, - = Last Active, * = Both
1::/64 *[Static/5] 00:01:50
> to 8:1::2 via ge-0/1/0.0
user@host# show route 2000::1
inet6.0: 26 destinations, 27 routes (25 active, 0 holddown, 1 hidden)
+ = Active Route, - = Last Active, * = Both
2000::/126 *[BGP/170] 00:05:32, MED 20, localpref 100
AS path: 2 I
> to 8:1::2 via ge-0/1/0.0
```

Configuring Aggregate Routes

Route aggregation allows you to combine groups of routes with common addresses into a single entry in the routing table. This decreases the size of the routing table as well as the number of route advertisements sent by the routing device.

An aggregate route becomes active when it has one or more *contributing routes*. A contributing route is an active route that is a more specific match for the aggregate destination. For example, for the aggregate destination 128.100.0.0/16, routes to 128.100.192.0/19 and 128.100.67.0/24 are contributing routes, but routes to 128.0.0.0./8, 128.0.0.0/16, and 128.100.0.0/16 are not.

A route can contribute only to a single aggregate route. However, an active aggregate route can recursively contribute to a less specific matching aggregate route. For example, an aggregate route to the destination 128.100.0.0/16 can contribute to an aggregate route to 128.96.0.0/13.

When an aggregate route becomes active, it is installed in the routing table with the following information:

- Reject next hop—If a more-specific packet does not match a more-specific route, the packet is rejected and an ICMP unreachable message is sent to the packet's originator.
- Metric value as configured with the aggregate statement.
- Preference value that results from the policy filter on the primary contributor, if a filter is specified.
- AS path as configured in the aggregate statement, if any. Otherwise, the path is computed by aggregating the paths of all contributing routes.
- Community as configured in the aggregate statement, if any is specified.

NOTE: You can configure only one aggregate route for each destination prefix.

To configure aggregate routes in the default routing table (inet.0), include the aggregate statement:

```
aggregate {
    defaults {
        ... aggregate-options ...
    }
    route destination-prefix {
        policy policy-name;
        ... aggregate-options ...
    }
}
```

To configure aggregate routes in one of the other routing tables, or to explicitly configure aggregate routes in the default routing table (inet.0), include the aggregate statement:

```
rib routing-table-name {
    aggregate {
        defaults {
            ... aggregate-options ...
        }
        route destination-prefix {
            policy policy-name;
            ... aggregate-options ...
        }
    }
}
```

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for these statements.

NOTE: You cannot configure aggregate routes for the IPv4 multicast routing table (inet.1) nor the IPv6 multicast routing table (inet6.1).

The aggregate statement consists of two parts:

- defaults—Here you specify global aggregate route options. These are treated as global defaults and apply to all the aggregate routes you configure in the aggregate statement. This part of the aggregate statement is optional.
- route—Here you configure individual aggregate routes. In this part of the aggregate statement, you optionally can configure aggregate route options. These options apply to the individual destination only and override any options you configured in the defaults part of the aggregate statement.

The following topics provide more information about configuring aggregate routes:

- Configuring the Destination of Aggregate Routes on page 89
- Configuring Aggregate Route Options on page 89
- Applying Policies to Aggregate Routes on page 94
- Advertising Aggregate Routes on page 95

Configuring the Destination of Aggregate Routes

When you configure an individual aggregate route in the route part of the aggregate statement, specify the destination of the route (in route *destination-prefix*) in one of the following ways:

- *network/mask-length*, where *network* is the network portion of the IP address and *mask-length* is the destination prefix length.
- default if this is the default route to the destination. This is equivalent to specifying an IP address of 0.0.0.0/0.

Configuring Aggregate Route Options

In the defaults and route parts of the aggregate statement, you can specify *aggregate-options*, which define additional information about aggregate routes that is included with the route when it is installed in the routing table. All aggregate options are optional. Aggregate options that you specify in the defaults part of the aggregate statement are treated as global defaults and apply to all the aggregate routes you configure in the aggregate statement. Aggregate options that you specify in the route part of the aggregate statement override any global aggregate options and apply to that destination only.

To configure aggregate route options, include one or more of them in the defaults or route part of the aggregate statement:

```
[edit]
routing-options {
```

```
aggregate {
   (defaults | route) {
      (active | passive);
      as-path <as-path> <origin (egp | igp | incomplete)> <atomic-aggregate>
         <aggregator as-number in-address>;
      community [ community-ids ];
      discard;
      (brief | full);
      (metric | metric2 | metric3 | metric4) metric <type type>;
      (preference | preference2 | color | color2) preference <type type>;
      tag string;
   }
}
}
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

The following sections explain how to configure aggregate route options:

- Configuring a Metric Value for Aggregate Routes on page 90
- Configuring a Preference Value for Aggregate Routes on page 90
- Configuring the Next Hop for Aggregate Routes on page 91
- Associating BGP Communities with Aggregate Routes on page 91
- Associating AS Paths with Aggregate Routes on page 92
- Including AS Numbers in Aggregate Route Paths on page 93
- Configuring an OSPF Tag String for Aggregate Routes on page 93
- Controlling Retention of Inactive Aggregate Routes in the Routing and Forwarding Tables on page 94

Configuring a Metric Value for Aggregate Routes

You can specify up to four metric values, starting with metric (for the first metric value) and continuing with metric2, metric3, and metric4 by including one or more of the following statements:

```
aggregate (defaults | route) {
   (metric | metric2 | metric3 | metric4) metric <type type>;
}
```

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for these statements.

In the type option, you can specify the type of route.

Configuring a Preference Value for Aggregate Routes

By default, aggregate routes have a preference value of 130. If the routing table contains a dynamic route to a destination that has a better (lower) preference value than this, the dynamic route is chosen as the active route and is installed in the forwarding table.

To modify the default preference value, specify a primary preference value (preference). You also can specify secondary preference value (preference2); and colors, which are even finer-grained preference values (color and color2). To do this, include one or more of the following statements:

```
aggregate (defaults | route) {
   (preference | preference2 | color | color2) preference <type type>;
}
```

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for these statements.

The preference value can be a number in the range from 0 through 4,294,967,295 ($2^{32} - 1$) with a lower number indicating a more preferred route. For more information about preference values, see "Route Preferences Overview" on page 6.

In the type option, you can specify the type of route.

Configuring the Next Hop for Aggregate Routes

By default, when aggregate routes are installed in the routing table, the next hop is configured as a reject route. That is, the packet is rejected and an ICMP unreachable message is sent to the packet's originator.

When you configure an individual route in the route part of the aggregate statement, or when you configure the defaults for aggregate routes, you can specify a discard next hop. This means that if a more specific packet does not match a more specific route, the packet is rejected and a reject route for this destination is installed in the routing table, but ICMP unreachable messages are not sent.

Being able to discard next hops allows you to originate a summary route, which can be advertised through dynamic routing protocols, and allows you to discard received traffic that does not match a more specific route than the summary route. To discard next hops, include the discard option:

```
discard;
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

Associating BGP Communities with Aggregate Routes

By default, no BGP community information is associated with aggregate routes. To associate community information with the routes, include the community option:

```
aggregate (defaults | route) {
   community [ community-ids ];
}
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement. *community-value* is the community identifier and can be a number in the range from 0 through 65,535.

community-ids is one or more community identifiers for either communities or extended communities.

The format for community identifiers is:

as-number:*community-value*

as-number is the AS number and can be a value in the range from 1 through 65,534.

You also can specify *community-ids* for communities as one of the following well-known community names, which are defined in RFC 1997:

- no-export—Routes containing this community name are not advertised outside a BGP confederation boundary.
- no-advertise—Routes containing this community name are not advertised to other BGP peers.
- no-export-subconfed—Routes containing this community name are not advertised to external BGP peers, including peers in other members' ASs inside a BGP confederation.

You can explicitly exclude BGP community information with an aggregate route using the none option. Include none when configuring an individual route in the route portion of the aggregate statement to override a community option specified in the defaults portion of the statement.

NOTE: Extended community attributes are not supported at the [edit routing-options] hierarchy level. You must configure extended communities at the [edit policy-options] hierarchy level. For information about configuring extended communities information, see the "Configuring the Extended Communities Attribute" section in the *JUNOS Policy Framework Configuration Guide*.

Associating AS Paths with Aggregate Routes

By default, the AS path for aggregate routes is built from the component routes. To manually specify the AS path and associate AS path information with the routes, include the as-path option:

```
aggregate (defaults | route) {
   as-path <as-path> <origin (egp | igp | incomplete)> <atomic-aggregate> <aggregator
      as-number in-address>;
}
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

as-path is the AS path to include with the route. It can include a combination of individual AS path numbers and AS sets. Enclose sets in brackets ([ ]). The first AS number in the path represents the AS immediately adjacent to the local AS. Each subsequent number represents an AS that is progressively farther from the local AS, heading toward the origin of the path.

NOTE: In JUNOS Release 9.1 and later, the numeric AS range is extended to provide BGP support for 4-byte AS numbers, as defined in RFC 4893, *BGP Support for Four-octet AS Number Space*. For the AS number, you can configure a value from 1 through 4,294,967,295.

You also can specify the AS path using the BGP origin attribute, which indicates the origin of the AS path information:

- egp—Path information originated in another AS.
- igp—Path information originated within the local AS.
- incomplete—Path information was learned by some other means.

To attach the BGP ATOMIC_AGGREGATE path attribute to the aggregate route, specify the atomic-aggregate option. This path attribute indicates that the local system selected a less specific route rather than a more specific route.

To attach the BGP AGGREGATOR path attribute to the aggregate route, specify the aggregator option. When using this option, you must specify the last AS number that formed the aggregate route (encoded as two octets), followed by the IP address of the BGP system that formed the aggregate route.

Including AS Numbers in Aggregate Route Paths

By default, all AS numbers from all contributing paths are included in the aggregate route's path. To include only the longest common leading sequences from the contributing AS paths, include the brief option when configuring the route. If doing this results in AS numbers being omitted from the aggregate route, the BGP ATOMIC_ATTRIBUTE path attribute is included with the aggregate route.

```
aggregate (defaults | route) {
   brief;
}
```

To explicitly have all AS numbers from all contributing paths be included in the aggregate route's path, include the full option when configuring routes. Include this option when configuring an individual route in the route portion of the aggregate statement to override a retain option specified in the defaults portion of the statement.

```
aggregate (defaults | route) {
   full;
}
```

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for these statements.

Configuring an OSPF Tag String for Aggregate Routes

By default, no OSPF tag strings are associated with aggregate routes. You can specify an OSPF tag string by including the tag option:

```
aggregate (defaults | route) {
   tag string;
}
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

Controlling Retention of Inactive Aggregate Routes in the Routing and Forwarding Tables

Static routes are only removed from the routing table if the next hop becomes unreachable, which happens if there are no contributing routes. To have an aggregate route remain continually installed in the routing and forwarding tables, include the passive option when configuring the route:

```
aggregate (defaults | route) {
   passive;
}
```

Routes that have been configured to remain continually installed in the routing and forwarding tables are marked with reject next hops when they are inactive.

To explicitly remove aggregate routes when they become inactive, include the active option when configuring routes. Include this option when configuring an individual route in the route portion of the aggregate statement to override a retain option specified in the defaults portion of the statement.

```
aggregate (defaults | route) {
   active;
}
```

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for these statements.

Applying Policies to Aggregate Routes

You can associate a routing policy when configuring an aggregate route's destination prefix in the routes part of the aggregate statement. Doing so provides the equivalent of an import routing policy filter for the destination prefix. That is, each potential contributor to an aggregate route, along with any aggregate options, is passed through the policy filter. The policy then can accept or reject the route as a contributor to the aggregate route and, if the contributor is accepted, the policy can modify the default preferences.

The following algorithm is used to compare two aggregate contributing routes in order to determine which one is the primary or preferred contributor:

1. Compare the protocol's preferences of the contributing routes. The lower the preference, the better the route. This is similar to the comparison that is done while determining the best route for the routing table.
2. Compare the protocol's preferences2 of the contributing routes. The lower preference2 value is better. If only one route has preferences2, then this route is preferred.

3. The preference values are the same. Proceed with a numerical comparison of the prefix values.
   a. The primary contributor is the numerically smallest prefix value.
   b. If the two prefixes are numerically equal, the primary contributor is the route that has the smallest prefix length value.
4. At this point, the two routes are the same. The primary contributor does not change. An additional next hop is available for the existing primary contributor.

A rejected contributor still can contribute to a less specific aggregate route. If you do not specify a policy filter, all candidate routes contribute to an aggregate route.

To associate a routing policy with an aggregate route, include the policy statement when configuring the route:

```
aggregate (defaults | route) {
   policy policy-name;
}
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

Advertising Aggregate Routes

After you have configured aggregate routes, you can have a protocol advertise the routes by configuring a policy that is then exported by a routing protocol.

To configure a protocol to advertise routes, include the policy-statement statement:

```
policy-statement advertise-aggregate-routes {
   term first-term {
      from protocol aggregate;
      then accept;
   }
   term second-term {
      then next policy;
   }
}
protocols {
   bgp {
      export advertise-aggregate-routes;
      ...
   }
}
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

Configuring Generated Routes

Generated routes are used as the *route of last resort*. A packet is forwarded to the route of last resort when the routing tables have no information about how to reach that packet's destination. One use of route generation is to generate a default route to use if the routing table contains a route from a peer on a neighboring backbone.

A generated route becomes active when it has one or more *contributing routes*. A contributing route is an active route that is a more specific match for the generated destination. For example, for the destination 128.100.0.0/16, routes to 128.100.192.0/19 and 128.100.67.0/24 are contributing routes, but routes to 128.0.0.0./8, 128.0.0.0/16, and 128.100.0.0/16 are not.

A route can contribute only to a single generated route. However, an active generated route can recursively contribute to a less specific matching generated route. For example, a generated route to the destination 128.100.0.0/16 can contribute to a generated route to 128.96.0.0/13.

By default, when generated routes are installed in the routing table, the next hop is chosen from the primary contributing route.

NOTE: Currently, you can configure only one generated route for each destination prefix.

To configure generated routes in the default routing table (inet.0), include the generate statement:

```
generate {
    defaults {
        generate-options;
    }
    route destination-prefix {
        policy policy-name;
        generate-options;
    }
}
```

To configure generated routes in one of the other routing tables, or to explicitly configure generated routes in the default route table (inet.0), include the generate statement:

```
rib routing-table-name {
    generate {
        defaults {
            generate-options;
        }
        route destination-prefix {
            policy policy-name;
            generate-options;
        }
    }
```

```
}
```

NOTE: You cannot configure generated routes for the IPv4 multicast routing table (inet.1) or the IPv6 multicast routing table (inet6.1).

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for these statements.

The generate statement consists of two parts:

- defaults—Here you specify global generated route options. These are treated as global defaults and apply to all the generated routes you configure in the generate statement. This part of the generate statement is optional.
- route—Here you configure individual generated routes. In this part of the generate statement, you optionally can configure generated route options. These options apply to the individual destination only and override any options you configured in the defaults part of the generate statement.

The following topics provide more information about configuring generated routes:

- Configuring the Destination of Generated Routes on page 97
- Configuring Generated Route Options on page 97
- Applying Policies to Generated Routes on page 102

Configuring the Destination of Generated Routes

When you configure an individual generated route in the route part of the generate statement, specify the destination of the route (in route *destination-prefix*) in one of the following ways:

- *network/mask-length*, where *network* i s the network portion of the IP address and *mask-length* is the destination prefix length.
- default if this is the default route to the destination. This is equivalent to specifying an IP address of 0.0.0.0/0.

Configuring Generated Route Options

In the defaults and route parts of the generate statement, you can specify options that define additional information about generated routes that is included with the route when it is installed in the routing table. All generated options are optional. Generated options that you specify in the defaults part of the generate statement are treated as global defaults and apply to all the generated routes you configure in the generate statement. Generated options that you specify in the route part of the generate statement override any global generated options and apply to that destination only.

To configure generated route options, include one or more of them in the defaults or route part of the generate statement (for routing instances, include the statement).

```
[edit]
routing-options
   generate {
      (defaults | route) {
         (active | passive);
         as-path <as-path> <origin (egp | igp | incomplete)> <atomic-aggregate>
            <aggregator as-number in-address>;
         community [ community-ids ];
         discard;
         (brief | full);
         (metric | metric2 | metric3 | metric4) metric <type type>;
         (preference | preference2 | color | color2) preference <type type>;
         tag string;
      }
   }
}
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

The following sections explain how to configure generated route options:

- Configuring a Metric Value for Generated Routes on page 98
- Configuring a Preference Value for Generated Routes on page 99
- Configuring the Next Hop for Generated Routes on page 99
- Associating BGP Communities with Generated Routes on page 99
- Associating AS Paths with Generated Routes on page 100
- Configuring an OSPF Tag String for Generated Routes on page 101
- Including AS Numbers in Generated Route Paths on page 101
- Controlling Retention of Inactive Generated Routes in the Routing and Forwarding Tables on page 102

Configuring a Metric Value for Generated Routes

You can specify up to four metric values, starting with metric (for the first metric value) and continuing with metric2, metric3, and metric4, by including one or more of the following statements:

```
(metric | metric2 | metric3 | metric4) metric < type type>;
```

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for these statements.

In the type option, you specify the type of route.

Configuring a Preference Value for Generated Routes

By default, generated routes have a preference value of 130. If the JUNOS routing table contains a dynamic route to a destination that has a better (lower) preference value than this, the dynamic route is chosen as the active route and is installed in the forwarding table.

To modify the default preference value, specify a primary preference value (preference). You also can specify a secondary preference value (preference2) and colors, which are even finer-grained preference values (color and color2). To do this, include one or more of the following statements:

(preference | preference2 | color | color2) *preference* <type *type*>;

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for these statements.

The preference value can be a number in the range from 0 through 4,294,967,295 ($2^{32} - 1$) with a lower number indicating a more preferred route. For more information about preference values, see "Route Preferences Overview" on page 6.

In the type option, you specify the type of route.

Configuring the Next Hop for Generated Routes

By default, when generated routes are installed in the routing table, the next hop is chosen from the primary contributing route.

When you configure an individual route in the route part of the generate statement, or when you configure the defaults for generated routes, you can specify a discard next hop. This means that if a more specific packet does not match a more specific route, the packet is rejected and a reject route for this destination is installed in the routing table, but ICMP unreachable messages are not sent. The discard next-hop feature allows you to originate a summary route, which can be advertised through dynamic routing protocols, and allows you to discard received traffic that does not match a more specific route than the summary route.

For example:

```
[edit routing-options generate route 1.0.0.0/8]
user@host# set discard
```

Associating BGP Communities with Generated Routes

By default, no BGP community information is associated with generated routes. To associate community information with the routes, include the community option:

community [*community-ids*];

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

community-ids is one or more community identifiers for either communities or extended communities.

The format for community identifiers is:

as-number:*community-value*

as-number is the AS number and can be a value in the range from 1 through 65,534.

You also can specify *community-ids* for communities as one of the following well-known community names, which are defined in RFC 1997:

- no-advertise—Routes containing this community name are not advertised to other BGP peers.
- no-export—Routes containing this community name are not advertised outside a BGP confederation boundary.
- no-export-subconfed—Routes containing this community name are not advertised to external BGP peers, including peers in other members' ASs inside a BGP confederation.

You can explicitly exclude BGP community information with a generated route using the none option. Include none when configuring an individual route in the route portion of the generate statement to override a community option specified in the defaults portion of the statement.

NOTE: Extended community attributes are not supported at the [edit routing-options] hierarchy level. You must configure extended communities at the [edit policy-options] hierarchy level. For information about configuring extended communities, see the "Configuring the Extended Communities Attribute" section in the *JUNOS Policy Framework Configuration Guide*.

Associating AS Paths with Generated Routes

By default, no AS path information is associated with generated routes. To associate AS path information with the routes, include the as-path statement:

as-path <*as-path*> <origin (egp | igp | incomplete)> <atomic-aggregate> <aggregator *as-number in-address*>;

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

as-path is the AS path to include with the route. It can include a combination of individual AS path numbers and AS sets. Enclose sets in brackets ([ ]). The first AS number in the path represents the AS immediately adjacent to the local AS. Each subsequent number represents an AS that is progressively farther from the local AS, heading toward the origin of the path.

NOTE: In JUNOS Release 9.1 and later, the numeric AS range is extended to provide BGP support for 4-byte AS numbers, as defined in RFC 4983, *BGP Support for Four-octet AS Number Space*. For the AS number, you can configure a number from 1 through 4,294,967,295.

You also can specify the AS path using the BGP origin attribute, which indicates the origin of the AS path information:

- egp—Path information originated in another AS.
- igp—Path information originated within the local AS.
- incomplete—Path information was learned by some other means.

To attach the BGP ATOMIC_AGGREGATE path attribute to the generated route, specify the atomic-aggregate option. This path attribute indicates that the local system selected a less specific route rather than a more specific route.

To attach the BGP AGGREGATOR path attribute to the generated route, specify the aggregator option. When using this option, you must specify the last AS number that formed the generated route (encoded as two octets), followed by the IP address of the BGP system that formed the generated route.

Configuring an OSPF Tag String for Generated Routes

By default, no OSPF tag strings are associated with generated routes. You can specify an OSPF tag string by including the tag statement:

```
tag string;
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

Including AS Numbers in Generated Route Paths

By default, all AS numbers from all contributing paths are included in the generated route's path. To include only the longest common leading sequences from the contributing AS paths, include the brief statement when configuring the route. If doing this results in AS numbers being omitted from the generated route, the BGP ATOMIC_ATTRIBUTE path attribute is included with the generated route.

```
brief;
```

To explicitly have all AS numbers from all contributing paths be included in the generated route's path, include the full state when configuring routes. Include this option when configuring an individual route in the route portion of the generate statement to override a retain option specified in the defaults portion of the statement.

```
full;
```

For a list of hierarchy levels at which you can include the brief or full statement, see the statement summary sections for these statements.

Controlling Retention of Inactive Generated Routes in the Routing and Forwarding Tables

Static routes are only removed from the routing table if the next hop becomes unreachable, which happens if there are no contributing routes. To have a generated route remain continually installed in the routing and forwarding tables, include the passive option when configuring the route:

```
generate (defaults | route) {
    passive;
}
```

Routes that have been configured to remain continually installed in the routing and forwarding tables are marked with reject next hops when they are inactive.

To explicitly remove generated routes when they become inactive, include the active option when configuring routes. Include this option when configuring an individual route in the route portion of the generate statement to override a retain option specified in the defaults portion of the statement.

```
generate (defaults | route) {
    active;
}
```

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for these statements.

Applying Policies to Generated Routes

You optionally can associate a routing policy when configuring a generated route's destination prefix in the routes part of the generate statement. Doing so provides the equivalent of an import routing policy filter for the destination prefix. That is, each potential contributor to a generated route, along with any generate options, is passed through the policy filter. The policy can accept or reject the route as a contributor to the generated route and, if the contributor is accepted, the policy can modify the default preferences.

The following algorithm is used to compare two generated contributing routes in order to determine which one is the primary or preferred contributor:

1. Compare the protocol's preference of the contributing routes. The lower the preference, the better the route. This is similar to the comparison that is done while determining the best route for the routing table.
2. Compare the protocol's preference2 of the contributing routes. The lower preference2 value is better. If only one route has preference2, then this route is preferred.
3. The preference values are the same. Proceed with a numerical comparison of the prefixes' values.
    a. The primary contributor is the numerically smallest prefix value.
    b. If the two prefixes are numerically equal, the primary contributor is the route that has the smallest prefix length value.

At this point, the two routes are the same. The primary contributor does not change. An additional next hop is available for the existing primary contributor.

A rejected contributor still can contribute to less specific generated route. If you do not specify a policy filter, all candidate routes contribute to a generated route.

To associate a routing policy with an generated route, include the policy statement:

```
policy policy-name;
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

Configuring Martian Addresses

Martian addresses are host or network addresses about which all routing information is ignored. They commonly are sent by improperly configured systems on the network and have destination addresses that are obviously invalid.

In IPv4, the following are the default martian addresses:

- 0.0.0.0/8
- 127.0.0.0/8
- 128.0.0.0/16
- 191.255.0.0/16
- 192.0.0.0/24
- 223.255.255.0/24
- 240.0.0.0/4

In IPv6, the loopback address, the reserved and unassigned prefixes from RFC 2373, and the link-local unicast prefix are the default martian addresses.

The following sections explain how to configure martian routes:

- Adding Martian Addresses on page 103
- Deleting Martian Addresses on page 104
- Using Class E Addresses for Interface Addresses on page 105

Adding Martian Addresses

To add martian addresses to the list of default martian addresses in the default IPv4 routing table (inet.0), include the martians statement:

```
martians {
    destination-prefix match-type;
}
```

To add martian addresses to the list of default martian addresses in other routing tables, or to explicitly add martian addresses to the list of default martian addresses in the primary IPv6 routing table (inet6.0), include the martians statement:

```
rib inet6.0 {
    martians {
        destination-prefix match-type;
    }
}
```

To add martian addresses to the list of default martian addresses in any other routing tables, or to explicitly add martian addresses to the list of default martian addresses in the default routing table (inet.0), include the martians statement:

```
rib routing-table-name {
    martians {
        destination-prefix match-type;
    }
}
```

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for these statements.

In *destination-prefix*, specify the routing destination in one of the following ways:

- default—If this is the default route to the destination. This is equivalent to specifying the IP address 0.0.0.0/0.
- *network/mask-length*—*network* is the network portion of the IP address and *mask-length* is the destination prefix length.

In *match-type*, specify the type of match to apply to the destination prefix. For more information about match types, see the *JUNOS Policy Framework Configuration Guide*.

Deleting Martian Addresses

To delete a martian address from within a range of martian addresses, include the allow option in the martians statement. This option removes an exact prefix that is within a range of addresses that has been specified to be martian addresses.

To delete a martian address from the default routing table (inet.0), include the martians statement:

```
martians {
    destination-prefix match-type allow;
}
```

To delete a martian address from other routing tables, or to explicitly delete a martian address from the primary IPv6 routing table (inet6.0), include the martians statement:

```
rib inet6.0 {
    martians {
        destination-prefix match-type allow;
    }
}
```

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for these statements.

Using Class E Addresses for Interface Addresses

In JUNOS Release 9.6 and later, you can configure Class E addresses on interfaces. Class E addresses are treated like any other unicast address for the purpose of forwarding. To allow Class E addresses to be configured on interfaces, you must remove the Class E prefix from the list of martian addresses. To remove the Class E prefix from the list of martian addresses include the martians 240/4 orlonger allow statement at the [edit routing-options] hierarchy level.

Configuring Flow Routes

A flow route is an aggregation of match conditions for IP packets. Flow routes are propagated through the network using flow-specification network-layer reachability information (NLRI) messages and installed into the flow routing table *instance-name*.inetflow.0. Packets can travel through flow routes only if specific match conditions are met.

Flow routes and firewall filters are similar in that they filter packets based on their components and perform an action on the packets that match. Flow routes provide traffic filtering and rate-limiting capabilities much like firewall filters. In addition, you can propagate flow routes across different autonomous systems.

To configure a flow route, include the flow statement:

```
flow {
    route name {
        match {
            match-conditions;
        }
        then {
            actions;
        }
    }
    term-order (legacy | standard);
    validation {
        traceoptions {
            file filename <files number> <size size> <world-readable | no-world-readable>;
            flag flag <flag-modifier> <disable>;
        }
    }
}
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

Flow routes are propagated by BGP through flow-specification NLRI messages. You must enable BGP to propagate these NLRIs. For more information on configuring BGP, see “BGP Configuration Guidelines” on page 721.

The following sections describe the specified tasks:

- Configuring Match Conditions for Flow Routes on page 106
- Configuring the Action for Flow Routes on page 108
- Validating Flow Routes on page 108
- Enabling Support for BGP Flow-Specification Algorithm Version 7 and Later on page 109

Configuring Match Conditions for Flow Routes

You specify conditions that the packet must match before the action in the **then** statement is taken for a flow route. All conditions in the **from** statement must match for the action to be taken. The order in which you specify match conditions is not important, because a packet must match all the conditions in a term for a match to occur.

To configure a match condition, include the **match** statement at the **[edit routing-options flow]** hierarchy level:

```
[edit routing-options flow]
match {
   match-conditions;
}
```

Table 4 on page 106 describes the flow route match conditions.

Table 4: Flow Route Match Conditions

Match Condition	Description
destination *prefix*	IP destination address field.
destination-port *number*	TCP or User Datagram Protocol (UDP) destination port field. You cannot specify both the **port** and **destination-port** match conditions in the same term. In place of the numeric value, you can specify one of the following text synonyms (the port numbers are also listed): **afs** (1483), **bgp** (179), **biff** (512), **bootpc** (68), **bootps** (67), **cmd** (514), **cvspserver** (2401), **dhcp** (67), **domain** (53), **eklogin** (2105), **ekshell** (2106), **exec** (512), **finger** (79), **ftp** (21), **ftp-data** (20), **http** (80), **https** (443), **ident** (113), **imap** (143), **kerberos-sec** (88), **klogin** (543), **kpasswd** (761), **krb-prop** (754), **krbupdate** (760), **kshell** (544), **ldap** (389), **login** (513), **mobileip-agent** (434), **mobilip-mn** (435), **msdp** (639), **netbios-dgm** (138), **netbios-ns** (137), **netbios-ssn** (139), **nfsd** (2049), **nntp** (119), **ntalk** (518), **ntp** (123), **pop3** (110), **pptp** (1723), **printer** (515), **radacct** (1813), **radius** (1812), **rip** (520), **rkinit** (2108), **smtp** (25), **snmp** (161), **snmptrap** (162), **snpp** (444), **socks** (1080), **ssh** (22), **sunrpc** (111), **syslog** (514), **tacacs-ds** (65), **talk** (517), **telnet** (23), **tftp** (69), **timed** (525), **who** (513), **xdmcp** (177), **zephyr-clt** (2103), or **zephyr-hm** (2104).
dscp *number*	Differentiated Services code point (DSCP). The DiffServ protocol uses the type-of-service (ToS) byte in the IP header. The most significant six bits of this byte form the DSCP. You can specify DSCP in hexadecimal or decimal form.

Table 4: Flow Route Match Conditions *(continued)*

Match Condition	Description
fragment *type*	Fragment type field. The keywords are grouped by the fragment type with which they are associated: ■ dont-fragment ■ first-fragment ■ is-fragment ■ last-fragment ■ not-a-fragment
icmp-code *number*	ICMP code field. This value or keyword provides more specific information than icmp-type. Because the value's meaning depends upon the associated icmp-type value, you must specify icmp-type along with icmp-code. In place of the numeric value, you can specify one of the following text synonyms (the field values are also listed). The keywords are grouped by the ICMP type with which they are associated: ■ parameter-problem: ip-header-bad (0), required-option-missing (1) ■ redirect: redirect-for-host (1), redirect-for-network (0), redirect-for-tos-and-host (3), redirect-for-tos-and-net (2) ■ time-exceeded: ttl-eq-zero-during-reassembly (1), ttl-eq-zero-during-transit (0) ■ unreachable: communication-prohibited-by-filtering (13), destination-host-prohibited (10), destination-host-unknown (7), destination-network-prohibited (9), destination-network-unknown (6), fragmentation-needed (4), host-precedence-violation (14), host-unreachable (1), host-unreachable-for-TOS (12), network-unreachable (0), network-unreachable-for-TOS (11), port-unreachable (3), precedence-cutoff-in-effect (15), protocol-unreachable (2), source-host-isolated (8), source-route-failed (5)
icmp-type *number*	ICMP packet type field. Normally, you specify this match in conjunction with the protocol match statement to determine which protocol is being used on the port. In place of the numeric value, you can specify one of the following text synonyms (the field values are also listed): echo-reply (0), echo-request (8), info-reply (16), info-request (15), mask-request (17), mask-reply (18), parameter-problem (12), redirect (5), router-advertisement (9), router-solicit (10), source-quench (4), time-exceeded (11), timestamp (13), timestamp-reply (14), or unreachable (3).
packet-length *number*	Total IP packet length.
port *number*	TCP or UDP source or destination port field. You cannot specify both the port match and either the destination-port or source-port match condition in the same term. In place of the numeric value, you can specify one of the text synonyms listed under destination-port.
protocol *number*	IP protocol field. In place of the numeric value, you can specify one of the following text synonyms (the field values are also listed): ah, egp (8), esp (50), gre (47), icmp (1), igmp (2), ipip (4), ipv6 (41), ospf (89), pim (103), rsvp (46), tcp (6), or udp (17).
source *prefix*	IP source address field.
source-port *number*	TCP or UDP source port field. You cannot specify the port and source-port match conditions in the same term. In place of the numeric field, you can specify one of the text synonyms listed under destination-port.
tcp-flag type	TCP header format.

Configuring the Action for Flow Routes

You can specify the action to take if the packet matches the conditions you have configured in the flow route. To configure an action, include the then statement at the [edit routing-options flow] hierarchy level:

```
[edit routing-options flow]
then {
   action;
}
```

Table 5 on page 108 describes the flow route actions.

Table 5: Flow Route Action Modifiers

Action or Action Modifier	Description
Actions	
accept	Accept a packet. This is the default.
discard	Discard a packet silently, without sending an Internet Control Message Protocol (ICMP) message.
community	Replace any communities in the route with the specified communities.
next-term	Continue to the next match condition for evaluation.
routing-instance *extended-community*	Specify a routing instance to which packets are forwarded.
rate-limit *rate*	Limit the bandwidth on the flow route.
sample	Sample the traffic on the flow route.

Validating Flow Routes

The JUNOS Software installs flow routes into the flow routing table only if they have been validated using the validation procedure. The Routing Engine does the validation before the installing routes into the flow routing table.

Flow routes received using the BGP network layer reachability information (NLRI) messages are validated before they are installed into the flow primary instance routing table instance.inetflow.0. The validation procedure is described in the draft-ietf-idr-flow-spec-09.txt, *Dissemination of Flow Specification Rules*. You can bypass the validation process for flow routes using BGP NLRI messages and use your own specific import policy.

To trace validation operations, include the validation statement at the [edit routing-options flow] hierarchy level:

```
[edit routing-options flow]
validation {
```

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

Enabling Support for BGP Flow-Specification Algorithm Version 7 and Later

By default, the JUNOS Software uses the term-ordering algorithm defined in version 6 of the BGP flow specification draft. In JUNOS Release 10.0 and later, you can configure the router to comply with the term-ordering algorithm first defined in version 7 of the BGP flow specification and supported through the latest version, Internet draft draft-ietf-idr-flow-spec-09.txt, *Dissemination of Flow Specification Routes*.

BEST PRACTICE: We recommend that you configure the JUNOS Software to use the term-ordering algorithm first defined in version 7 of the BGP flow specification draft. We also recommend that you configure the JUNOS Software to use the same term-ordering algorithm on all routing instances configured on a router.

To configure BGP to use the flow-specification algorithm first defined in version 7 of the Internet drat, Include the standard statement at the [edit routing-options flow term-order] hierarchy level:

```
[edit routing-options]
flow {
    term-order standard;
}
```

To revert to using the term-ordering algorithm defined in version 6, include the legacy statement at the [edit routing-options flow term-order] hierarchy level:

```
[edit routing-options]
flow {
    term-order legacy;
}
```

NOTE: The configured term order has only local significance. That is, the term order does not propagate with flow routes sent to the remote BGP peers, whose term order is completely determined by their own term order configuration. Therefore, you should be careful when configuring the order-dependent action next term when you are not aware of the term order configuration of the remote peers. The local next term might differ from the next term configured on the remote peer.

Related Topics flow

Applying Filters to the Forwarding Table

To apply an input filter to a forwarding table, include the input statement:

```
rib routing-table-name {
    filter {
        input filter-name;
    }
}
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

NOTE: Forwarding table filtering is not supported on the interfaces you configure as tunnel sources. Input filters affect only the transit packets exiting the tunnel.

Forwarding table filtering is not supported with the flow routes configuration.

For more information about forwarding table filters, see the *JUNOS Policy Framework Configuration Guide*.

Chapter 5

Configuring Other Protocol-Independent Routing Properties

This chapter discusses how to perform the following tasks for configuring other protocol-independent routing properties:

- Configuring AS Numbers for BGP on page 112
- Configuring Router Identifiers for BGP and OSPF on page 113
- Configuring AS Confederation Members on page 113
- Configuring Route Recording for Flow Aggregation on page 114
- Creating Routing Table Groups on page 114
- Configuring How Interface Routes Are Imported into Routing Tables on page 116
- Configuring Multicast Scoping on page 117
- Enabling Multicast Forwarding Without PIM on page 118
- Configuring Additional Source-Specific Multicast Groups on page 119
- Configuring Multicast Forwarding Cache Limits on page 119
- Configuring Per-Packet Load Balancing on page 120
- Configuring Unicast Reverse-Path-Forwarding Check on page 122
- Configuring Graceful Restart on page 124
- Configuring Route Distinguishers for VRF and Layer 2 VPN Instances on page 125
- Configuring Dynamic GRE Tunnels for VPNs on page 125
- Configuring System Logging for the Routing Protocol Process on page 126
- Configuring Route Resolution on page 127
- Enabling Indirect Next Hops on page 128
- Enabling Nonstop Active Routing on page 129
- Tracing Global Routing Protocol Operations on page 129
- Disabling Distributed Periodic Packet Management on the Packet Forwarding Engine on page 131
- Enabling Source Routing on page 132
- Delaying Updates of the MED Path Attribute for BGP on page 133
- Creating Policies to Control Label Allocation and Substitution for MPLS Ingress and AS Border Routers on page 133

Configuring AS Numbers for BGP

An autonomous system (AS) is a set of routing devices that are under a single technical administration and that generally use a single interior gateway protocol (IGP) and metrics to propagate routing information within the set of routing devices. An AS appears to other ASs to have a single, coherent interior routing plan and presents a consistent picture of what destinations are reachable through it.

ASs are identified by a number that is assigned by the Network Information Center (NIC) in the United States (http://www.isi.edu). In JUNOS Release 9.1 and later, you can configure a number from 1 through 4,294,967,295 in plain-number format. The range is extended to provide BGP support for 4-byte AS numbers, as defined in RFC 4893, *BGP Support for Four-octet AS Number Space*. All releases of the JUNOS Software support 2-byte AS numbers.

In JUNOS Release 9.3 and later, you can also configure a 4-byte AS number using the AS-dot notation format of two integer values joined by a period: *< 16-bit high-order value in decimal >* . *< 16-bit low-order value in decimal >* . For example, the 4-byte AS number of 65,546 in plain-number format is represented as 1.10 in the AS-dot notation format. In AS-dot notation format, you can specify a value for AS number from 0.0 through 65535.65535.

If you are using BGP on the routing device, you must configure an AS number.

To configure the routing device's AS number, include the autonomous-system statement:

autonomous-system *autonomous-system* <asdot-notation> <loops *number*>;

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

To specify the maximum number of times that this AS number can appear in an AS path, include the loops option. You can specify a value in the range from 1 through 10. The default value is 1.

The AS path attribute is modified when a route is advertised to an EBGP peer. Each time a route is advertised to an EBGP peer, the local routing device prepends its AS number to the existing path attribute, and a value of 1 is added to the AS number. The default loop value of 1 means that an AS number can appear in an AS path only one time. That is, when the local routing device advertises an AS path to an EBGP peer, that peer cannot advertise that AS path to another EBGP peer. To ensure that the AS path can be advertised by the peer that receives the route to another EBGP peer, specify a loops value of 2.

NOTE: When you specify the same AS number in more than one routing instance on the local routing device, you must configure the same number of loops for the AS number in each instance. For example, if you configure a value of 3 for the loops statement in a VPN routing and forwarding (VRF) routing instance that uses the same AS number as that of the master instance, you must also configure a value of 3 loops for the AS number in the master instance.

Use the independent-domain option if the loops statement must be enabled only on a subset of routing instances. For more information about configuring an independent AS domain, see "Configuring Independent AS Domains" on page 271.

By default, the AS number is displayed in plain-number format even if you configured a 4-byte AS number using the AS-dot notation format. Include the asdot-notation statement to configure the router to display a 4-byte AS number in the AS-dot notation format.

Configuring Router Identifiers for BGP and OSPF

The router identifier is used by BGP and OSPF to identify the routing device from which a packet originated. The router identifier usually is the IP address of the local routing device. If you do not configure a router identifier, the IP address of the first interface to come online is used. This is usually the loopback interface. Otherwise, the first hardware interface with an IP address is used.

To configure the router identifier, include the router-id statement:

```
router-id address;
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

NOTE: We strongly recommend that you configure the router identifier under the [edit routing-options] hierarchy level to avoid unpredictable behavior if the interface address on a loopback interface changes.

Configuring AS Confederation Members

If you administer multiple ASs that contain a very large number of BGP systems, you can group them into one or more *confederations*. Each confederation is identified by its own AS number, which is called a *confederation AS number*. To external ASs, a confederation appears to be a single AS. Thus, the internal topology of the ASs making up the confederation is hidden.

The BGP path attributes NEXT_HOP, LOCAL_PREF, and MULTI_EXIT_DISC, which normally are restricted to a single AS, are allowed to be propagated throughout the ASs that are members of the same confederation.

Because each confederation is treated as if it were a single AS, you can apply the same routing policy to all the ASs that make up the confederation.

Grouping ASs into confederations reduces the number of BGP connections required to interconnect ASs.

If you are using BGP, you can enable the local routing device to participate as a member of an AS confederation. To do this, include the confederation statement:

```
confederation confederation-autonomous-system members [ autonomous-systems ];
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

Specify the AS confederation identifier, along with the AS numbers that are members of the confederation.

Note that peer adjacencies do not form if two BGP neighbors disagree about whether an adjacency falls within a particular confederation.

Configuring Route Recording for Flow Aggregation

Before you can perform flow aggregation, the routing protocol process must export the AS path and routing information to the sampling process. To do this, include the route-record statement:

```
route-record;
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

For more information about flow aggregation and sampling, see the *JUNOS Network Interfaces Configuration Guide*.

Creating Routing Table Groups

You can group together one or more routing tables to form a *routing table group*. Within a group, a routing protocol can import routes into all the routing tables in the group and can export routes from a single routing table.

To create a routing table group, include the rib-groups statement:

```
rib-groups group-name {
    import-policy [ policy-names ];
    import-rib [ routing-table-names ];
    export-rib routing-table-name;
}
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

The routing table group can have any name you choose (specified in *group-name*). If the group name you specify is not created explicitly, as described in "Configuring

Other Protocol-Independent Routing Properties" on page 111, you can create it by naming it in the rib-groups statement.

Each routing table group must contain one or more routing tables that the JUNOS Software uses when importing routes (specified in the import-rib statement). The first routing table you specify is the *primary routing table*, and any additional routing tables are the *secondary routing tables*.

The primary routing table determines the address family of the routing table group. To configure an IP version 4 (IPv4) routing table group, specify inet.0 as the primary routing table. To configure an IP version 6 (IPv6) routing table group, specify inet6.0 as the primary routing table. If you configure an IPv6 routing table group, the primary and all secondary routing tables must be IPv6 routing tables (inet6.*x*).

In JUNOS Release 9.5 and later, you can include both IPv4 and IPv6 routing tables in an IPv4 import routing table group using the import-rib statement. In releases prior to JUNOS Release 9.5, you can only include either IPv4 or IPv6 routing tables in the same import-rib statement. The ability to configure an import routing table group with both IPv4 and IPv6 routing tables enables you, for example, to populate the inet6.3 routing table with IPv6 addresses that are compatible with IPv4. Specify inet.0 as the primary routing table, and specify inet6.3 as a secondary routing table.

Each routing table group optionally can contain one routing table group that the JUNOS Software uses when exporting routes to the routing protocols (specified in the export-rib statement).

NOTE: If you configure an import routing table group that includes both IPv4 and IPv6 routing tables, any corresponding export routing table group must include only IPv4 routing tables.

If you have configured a routing table, configure the OSPF primary instance at the [edit protocols ospf] hierarchy level with the statements needed for your network so that routes are installed in inet.0 and in the forwarding table. Make sure to include the routing table group. For more information, see "Configuring Multiple Instances of OSPF" on page 247.

After specifying the routing table from which to import routes, you can apply one or more policies to control which routes are installed in the routing table group. To apply a policy to routes being imported into the routing table group, include the import-policy statement:

```
rib-groups group-name {
    import-policy [ policy-names ];
}
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

Examples: Creating Routing Table Groups

Create an IPv4 routing table group so that interface routes are installed into two routing tables, inet.0 and inet.2:

```
[edit]
routing-options {
    interface-routes {
        rib-group if-rg;
    }
    rib-groups if-rg {
        import-rib [ inet.0 inet.2 ];
    }
}
```

Create an IPv6 routing table group so that interface routes are installed into two routing tables, inet6.0 and inet6.2:

```
[edit]
routing-options {
    interface-routes {
        rib-group inet6 if-rg;
    }
    rib-groups if-rg {
        import-rib [ inet6.0 inet6.2 ];
    }
}
```

Configuring How Interface Routes Are Imported into Routing Tables

By default, IPv4 interface routes (also called direct routes) are imported into routing table inet.0, and IPv6 interface routes are imported into routing table inet6.0. If you are configuring alternate routing tables for use by some routing protocols, it might be necessary to import the interface routes into the alternate routing tables. To define the routing tables into which interface routes are imported, you create a routing table group and associate it with the routing device's interfaces.

To associate an IPv4 routing table group with the routing device's interfaces and specify which routing table groups interface routes are imported into, include the interface-routes statement:

```
interface-routes {
    rib-group group-name;
}
```

To associate an IPv6 routing table group with an interface, include the interface-routes statement at the [edit routing-options] hierarchy level:

```
interface-routes {
    rib-group inet6 group-name;
}
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

To create the routing table groups, include the passive statement at the [edit routing-options] hierarchy level. For configuration information, see "Creating Routing Table Groups" on page 114.

If you have configured a routing table, configure the OSPF primary instance at the [edit protocols ospf] hierarchy level with the statements needed for your network so that routes are installed in inet.0 and in the forwarding table. Make sure to include the routing table group. For more information, see "Configuring Multiple Instances of OSPF" on page 247.

To export local routes, include the export statement:

```
export {
    lan;
    point-to-point;
}
```

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for these statements.

To export LAN routes, include the lan option. To export point-to-point routes, include the point-to-point option.

Only local routes on point-to-point interfaces configured with a destination address are exportable.

Configuring Multicast Scoping

To configure multicast address scoping, include the following statements:

```
multicast {
    scope scope-name {
        interface [ interface-names ];
        prefix destination-prefix;
    }
    scope-policy policy-name;
}
```

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for these statements.

Specify a name for the scope, its address range, and the routing device interfaces on which you are configuring scoping.

You can apply a multicast scoping policy to the routing table. To apply a scoping policy, include the scope-policy statement at the [edit routing-options multicast] hierarchy level. For more information on configuring a scoping policy, see the *JUNOS Policy Framework Configuration Guide*.

Example: Configuring Multicast Scoping

Configure multicast scoping by creating four scopes: local, organization, engineering, and marketing.

Configure the local scope on a Fast Ethernet interface. Configure the organization scope on a Fast Ethernet and a SONET/SDH interface. Configure the engineering and marketing scopes on two SONET/SDH interfaces.

```
[edit]
routing-options {
   multicast {
      scope local {
         prefix 239.255.0.0/16;
         fe-0/1/0.0;
      }
      scope organization {
         prefix 239.192.0.0/14;
         interface [ fe-0/1/0.0 so-0/0/0.0 ];
      }
      scope engineering {
         prefix 239.255.255.0/24;
         interface [ so-0/0/1.0 so-0/0/2.0 ];
      }
      scope marketing {
         prefix 239.255.254.0/24;
         interface [ so-0/0/1.0 so-0/0/2.0 ];
      }
   }
}
```

Enabling Multicast Forwarding Without PIM

By default, multicast packets are forwarded by enabling Protocol Independent Multicast (PIM) on an interface. PIM adds multicast routes into the routing table.

You can also configure multicast packets to be forwarded over a static route, such as a static route associated with an LSP next hop. Multicast packets are accepted on an interface and forwarded over a static route in the forwarding table. This is useful when you want to enable multicast traffic on a specific interface without configuring PIM on the interface.

To enable multicast traffic on an interface, include the **interface** statement:

```
interface interface-name;
```

To disable multicast traffic on an interface, include the **disable** statement:

```
interface interface-name {
   disable;
}
```

For a list of hierarchy levels at which you can include these statements, see the statement summary section for these statements.

NOTE: You cannot enable multicast traffic on an interface and configure PIM on the same interface simultaneously.

NOTE: Static routes must be configured before you can enable multicast on an interface. Configuring the interface statement alone does not install any routes into the routing table. This feature relies on the static route configuration.

Configuring Additional Source-Specific Multicast Groups

IGMPv3 supports Source Specific Multicast (SSM) groups. By utilizing inclusion lists, only sources that are specified send to the SSM group. By default, the SSM group multicast address is limited to the IP address range 232.0.0.0 to 232.255.255.255. You can configure additional SSM groups. Shared tree delivery is prohibited on SSM groups.

To configure additional SSM groups, include the ssm-groups statement:

```
ssm-groups {
   address;
}
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

Configuring Multicast Forwarding Cache Limits

To configure multicast forwarding cache limits, include the following statements:

```
multicast {
   forwarding-cache {
      threshold suppress value <reuse value>;
   }
}
```

You can include these statements at the following hierarchy levels:

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

For an overview of logical systems and a detailed example of logical system configuration, see the logical systems chapter of the *JUNOS Feature Guide*.

By default, there are no limits on the number of multicast forwarding cache entries.

Specify a value for the threshold at which to suppress new multicast forwarding cache entries and an optional reuse value for the threshold at which the router begins to create new multicast forwarding cache entries. The range for both is from 1 through 200,000. If configured, the reuse value should be less than the suppression threshold value. The suppression value is mandatory. If you do not specify the optional reuse value, then the number of multicast forwarding cache entries is limited to the suppression value. A new entry is created as soon as the number of multicast forwarding cache entries falls below the suppression value.

For information about supported standards for multicast scoping, see the *JUNOS Multicast Protocols Configuration Guide*.

Configuring Per-Packet Load Balancing

For the active route, when there are multiple equal-cost paths to the same destination, by default, the JUNOS Software chooses in a random fashion one of the next-hop addresses to install into the forwarding table. Whenever the set of next hops for a destination changes in any way, the next-hop address is chosen again, also in a random fashion.

You can configure the JUNOS Software so that, for the active route, all next-hop addresses for a destination are installed in the forwarding table. This is called per-packet load balancing. You can use load balancing to spread traffic across multiple paths between routing devices. The behavior of the per-packet load-balancing function varies according to the version of the Internet Processor ASIC in the routing device.

On routing devices with an Internet Processor I ASIC, when per-packet load balancing is configured, traffic between routindg devices with multiple paths is spread in a random fashion across the available interfaces. The forwarding table balances the traffic headed to a destination, transmitting packets in round-robin fashion among the multiple next hops (up to a maximum of eight equal-cost load-balanced paths). The traffic is load-balanced on a per-packet basis.

NOTE: Per-packet load distribution uses a hashing algorithm that distributes packets over equal-cost links. The algorithm is designed to distribute packets to prevent any single link from being saturated. However, per-packet load balancing offers no guarantee of equal distribution of traffic over equal-cost links, nor does it guarantee that increasing the number of Internet flows creates a better hash distribution.

On routing devices with the Internet Processor II ASIC and T Series Internet Processor II ASIC, when per-packet load balancing is configured, traffic between routing devices with multiple paths is divided into individual traffic flows (up to a maximum of 16 equal-cost load-balanced paths). Packets for each individual flow are kept on a single interface. To recognize individual flows in the transit traffic, the routing device examines each of the following:

- Source IP address
- Destination IP address
- Protocol
- Source port number

- Destination port number
- Source interface index
- Type of service (ToS)

The routing device recognizes packets in which all of these parameters are identical, and it ensures that these packets are sent out through the same interface. This prevents problems that might otherwise occur with packets arriving at their destination out of their original sequence.

The following steps shows how to configure per-packet load balancing:

1. Define a load-balancing routing policy by including one or more policy-statement statements at the [edit policy-options] hierarchy level, defining an action of load-balance per-packet:

```
policy-statement policy-name {
   from {
      match-conditions;
      route-filter destination-prefix match-type <actions>;
      prefix-list name;
   }
   then {
      load-balance per-packet;
   }
}
```

2. Apply the policy to routes exported from the routing table to the forwarding table. To do this, include the forwarding-table and export statements:

```
forwarding-table {
   export policy-name;
}
```

NOTE: You cannot apply the export policy to VRF routing instances.

For a list of hierarchy levels at which you can include these statements, see the statement summary section for these statements.

NOTE: Specify all next-hops of that route, if more than one exists, when allocating a label corresponding to a route that is being advertised.

NOTE: Configure the forwarding-options hash key for MPLS to include the IP payload.

Examples: Configuring Per-Packet Load Balancing

Perform per-packet load balancing for all routes:

```
[edit]
policy-options {
   policy-statement load-balancing-policy {
      then {
         load-balance per-packet;
      }
   }
}
routing-options {
   forwarding-table {
      export load-balancing-policy;
   }
}
```

Perform per-packet load balancing for only a limited set of routes:

```
[edit]
policy-options {
   policy-statement load-balancing-policy {
      from {
         route-filter 192.168.10/24 orlonger;
         route-filter 9.114/16 orlonger;
      }
      then {
         load-balance per-packet;
      }
   }
}
routing-options {
   forwarding-table {
      export load-balancing-policy;
   }
}
```

Configuring Unicast Reverse-Path-Forwarding Check

IP spoofing can occur during a denial-of-service (DoS) attack. IP spoofing allows an intruder to pass IP packets to a destination as genuine traffic, when in fact the packets are not actually meant for the destination. This type of spoofing is harmful because it consumes the destination's resources.

Unicast reverse-path-forwarding (RPF) check is a tool to reduce forwarding of IP packets that may be spoofing an address. A unicast RPF check performs a route table lookup on an IP packet's source address, and checks the incoming interface. The router determines whether the packet is arriving from a path that the sender would use to reach the destination. If the packet is from a valid path, the router forwards the packet to the destination address. If it is not from a valid path, the router discards the packet. Unicast RPF is supported for the IPv4 and IPv6 protocol families, as well as for the virtual private network (VPN) address family.

To control the operation of unicast RPF check, include the unicast-reverse-path statement:

```
unicast-reverse-path (active-paths | feasible-paths);
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

To consider only active paths during the unicast RPF check, include the active-paths option. To consider all feasible paths during the unicast RPF check, include the feasible-paths option.

NOTE: Reverse-path forwarding is not supported on the interfaces you configure as tunnel sources. This affects only the transit packets exiting the tunnel.

You must enable unicast RPF check on an interface. To do so, include the rpf-check statement:

```
rpf-check <fail-filter filter-name>;
```

You can include this statement at the following hierarchy levels:

- [edit interfaces *interface-name* unit *logical-unit-number* family (inet | inet6)]
- [edit logical-systems *logical-system-name* interfaces *interface-name* unit *logical-unit-number* family (inet | inet6)]

For more information about configuring unicast RPF on an interface, see the *JUNOS Network Interfaces Configuration Guide*.

Example: Configuring Unicast RPF

Configure unicast RPF strict mode, and apply a fail filter that allows the interface to accept BOOTP packets and DHCP packets. The filter accepts all packets with a source address of 0.0.0.0 and a destination address of 255.255.255.255.

```
[edit firewall]
filter rpf-special-case-dhcp-bootp {
    term allow-dhcp-bootp {
        from {
            source-address {
                0.0.0.0/32;
            }
            address {
                255.255.255.255/32;
            }
        }
        then {
            count rpf-dhcp-bootp-traffic;
            accept;
        }
    }
    term default {
        then {
            log;
            reject;
        }
    }
```

```
}
[edit]
interfaces {
    so-0/0/0 {
        unit 0 {
            family inet {
                rpf-check fail-filter rpf-special-case-dhcp-bootp;
            }
        }
    }
}
```

Configuring Graceful Restart

Graceful restart allows a routing device undergoing a restart to inform its adjacent neighbors and peers of its condition. The restarting routing device requests a grace period from the neighbor or peer, which can then cooperate with the restarting routing device. With a graceful restart, the restarting routing device can still forward traffic during the restart period, and convergence in the network is not disrupted. The restart is not visible to the rest of the network, and the restarting routing device is not removed from the network topology.

The graceful restart request occurs only if the following conditions are met:

- The network topology is stable.
- The neighbor or peer cooperates.
- The restarting routing device is not already cooperating with another restart already in progress.
- The grace period does not expire.

Graceful restart is disabled by default. You must configure graceful restart at the [edit routing-options] hierarchy level to enable the feature for Layer 2 and Layer 3 VPNs.

To enable graceful restart, include the graceful-restart statement:

```
graceful-restart {
    disable;
    restart-duration seconds;
}
```

To disable graceful restart, include the disable statement. To configure a time period for complete restart, include the restart-duration statement. You can specify a number between 120 and 900.

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

For a detailed example of a graceful restart configuration, see the *JUNOS High Availability Configuration Guide*.

Configuring Route Distinguishers for VRF and Layer 2 VPN Instances

If the route distinguisher ID is configured, the routing process automatically generates a type 1 route distinguisher for VPN routing and forwarding (VRF) and Layer 2 VPN instances. If a route distinguisher is explicitly configured under the routing instances stanza, then that configured route distinguisher is used.

To configure a route distinguisher identifier globally, include the route-distinguisher-id statement:

```
route-distinguisher-id address;
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

For more information about VPNs, see the *JUNOS VPNs Configuration Guide*.

Configuring Dynamic GRE Tunnels for VPNs

A VPN that travels through a non-MPLS network requires a generic routing encapsulation (GRE) tunnel. This tunnel can be either a static tunnel or a dynamic tunnel. A static tunnel is configured manually between two provider edge (PE) routers. A dynamic tunnel is configured using BGP route resolution.

When a router receives a VPN route that resolves over a BGP next hop that does not have an MPLS path, a GRE tunnel can be created dynamically, allowing the VPN traffic to be forwarded to that route. Formerly, GRE tunnels had to be established manually. Only GRE IPv4 tunnels are supported.

To configure a dynamic tunnel between two PE routers, include the dynamic-tunnels statement:

```
dynamic-tunnels tunnel-name {
    destination-networks prefix;
    source-address address;
    tunnel-type type;
}
```

You can include this statement at the following hierarchy levels:

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

Specify the IPv4 prefix range (for example, 10/8 or 11.1/16) for the destination network by including the destination-networks statement. Only tunnels within the specified IPv4 prefix range can be created.

```
destination-networks prefix;
```

You can include this statement at the following hierarchy levels:

- [edit routing-options dynamic-tunnels *tunnel-name*]
- [edit logical-systems *logical-system-name* routing-options dynamic-tunnels *tunnel-name*]

Specify the source address for the GRE tunnels by including the source-address statement. The source address specifies the address used as the source for the local tunnel endpoint. It can be any local address on the router (typically the router ID or the loopback address).

```
source-address address;
```

You can include this statement at the following hierarchy levels:

- [edit routing-options dynamic-tunnels *tunnel-name*]
- [edit logical-systems *logical-system-name* routing-options dynamic-tunnels *tunnel-name*]

Specify the type of tunnel to be dynamically created by including the tunnel-type statement. The only currently valid value is gre (for GRE tunnels).

```
tunnel-type type;
```

You can include this statement at the following hierarchy levels:

- [edit routing-options dynamic-tunnels *tunnel-name*]
- [edit logical-systems *logical-system-name* routing-options dynamic-tunnels *tunnel-name*]

Configuring System Logging for the Routing Protocol Process

To control how much information the routing protocol process should log, include the options statement.

Include the following form of the statement to log messages for a particular severity level and all higher levels:

```
routing-options {
    options syslog upto level;
}
```

Include the following form of the statement to log messages for a particular severity level:

```
routing-options {
    options syslog level level;
}
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

NOTE: System logging frequently deals with processes logged at the info or notice severity level. Make sure that your regular system logging configurations include the info or notice levels.

Examples: Configuring System Logging for the Routing Protocol Process

Configure the router to log messages of all severities:

```
[edit]
user@host# set routing-options options syslog upto emergency
[edit]
user@host# show
routing-options {
   options syslog upto emergency;
}
```

Configure the router to log only alert-level and critical-level messages:

```
[edit]
user@host# set routing-options options syslog level alert critical
[edit]
user@host# show
routing-options {
   options syslog alert critical;
}
```

Configuring Route Resolution

You can configure a routing table to accept routes from specific routing tables. You can also configure a routing table to use specific import policies to produce a route resolution table to resolve routes.

To configure route resolution, include the resolution statement:

```
resolution {
   rib routing-table-name {
      import [ policy-names ];
      resolution-ribs [ routing-table-names ];
   }
}
```

To specify the name of the routing table to modify, include the rib *routing-table-name* statement. To specify one or more import policies to use for route resolution, include the import [*policy-names*] statement. To specify one or more routing tables to use for route resolution, include the resolution-ribs [*routing-table-names*] statement.

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

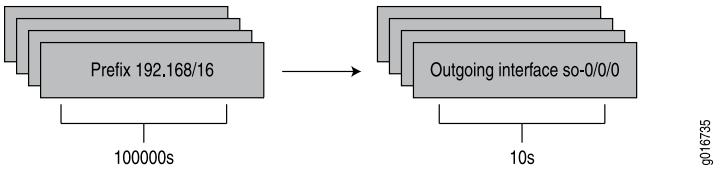
Enabling Indirect Next Hops

The JUNOS Software supports the concept of an indirect next hop for all routing protocols that support indirectly connected next hops, also known as third-party next hops.

Because routing protocols such as IBGP can send routing information about indirectly connected routes, the JUNOS Software relies on routes from intra-AS routing protocols (OSPF, IS-IS, RIP, and static) to resolve the best directly connected next hop. The Routing Engine performs the task of route resolution to determine the best directly connected next hop and install the route to the Packet Forwarding Engine.

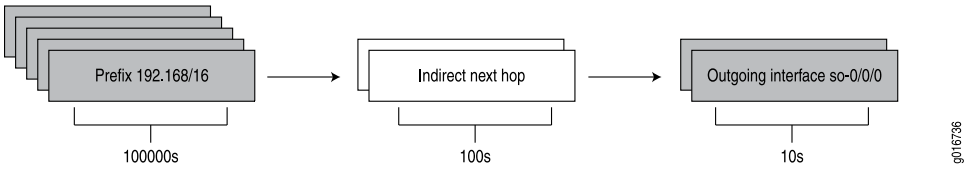
By default, the JUNOS Software does not maintain the route for indirect next hop to forwarding next-hop binding on the Packet Forwarding Engine forwarding table. As a result, when a rerouting event occurs, potentially thousands of route to forwarding next-hop bindings must be updated, which increases the route convergence time. Figure 2 on page 128 illustrates the route to forwarding next-hop bindings with indirect next hop disabled.

Figure 2: Route to Forwarding Next-Hop Bindings



You can enable the JUNOS Software to maintain the indirect next hop to forwarding next-hop binding on the Packet Forwarding Engine forwarding table. As a result, fewer route to forwarding next-hop bindings need to be updated, which improves the route convergence time. Figure 3 on page 128 illustrates the route to forwarding next-hop bindings with indirect next hop enabled.

Figure 3: Route to Forwarding Indirect Next-Hop Bindings



To enable indirectly connected next hops, include the indirect-next-hop statement:

```
indirect-next-hop;
```

NOTE: When virtual private LAN service (VPLS) is configured on the routing device, the indirect-next-hop statement is not supported at the [edit routing-options forwarding-table] hierarchy level.

To disable indirectly connected next hops, include the no-indirect-next-hop statement:

```
no-indirect-next-hop;
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

Enabling Nonstop Active Routing

Nonstop active routing (*NSR*) allows a routing platform with redundant Routing Engines to switch over from a primary Routing Engine to a backup Routing Engine without alerting peer nodes that a change has occurred. NSR uses the same infrastructure as graceful Routing Engine switchover (GRES) to preserve interface and kernel information. However, NSR also saves routing protocol information by running the routing protocol (rpd) process on the backup Routing Engine. Saving this additional information makes NSR self-contained and eliminates the need for helper routers to assist the routing platform in restoring routing protocol information. As a result of this enhanced functionality, NSR is a natural replacement for graceful restart protocol extensions.

If the kernel on the master Routing Engine stops operating, the master Routing Engine experiences a hardware failure, or the administrator initiates a manual switchover, mastership switches to the backup Routing Engine. To configure NSR, you must first enable GRES on your routing platform. For more information about how to configure GRES, see the *JUNOS High Availability Configuration Guide*.

To enable NSR, include the nonstop-routing statement at the [edit routing-options] hierarchy level.

```
[edit routing-options]
nonstop-routing;
```

NOTE: You cannot configure NSR and graceful restart protocol extensions simultaneously. To ensure proper operation, include either the nonstop-routing statement or the graceful-restart statement at the hierarchy level, but not both statements at the same time.

For more detailed information about NSR, see the *JUNOS High Availability Configuration Guide*.

Tracing Global Routing Protocol Operations

Global routing protocol tracing operations track all general routing operations and record them in a log file. To set protocol-specific tracing operations and to modify the global tracing operations for an individual protocol, configure tracing for that protocol.

For a general discussion about tracing and the precedence of multiple tracing operations, see the *JUNOS System Basics Configuration Guide*.

To configure global routing protocol tracing, include the traceoptions statement at the [edit routing-options] hierarchy level:

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

Using the traceoptions statement, you can specify the following global routing protocol tracing flags:

- all—All tracing operations
- condition-manager—Condition manager events
- config-internal—Configuration internals
- general—All normal operations and routing table changes (a combination of the normal and route trace operations)
- graceful-restart—Graceful restart operations
- normal—All normal operations
- nsr-synchronization—Nonstop-routing synchronization events
- parse—Configuration parsing
- policy—Policy operations and actions
- regex-parse—Regular-expression parsing
- route—Routing table changes
- state—State transitions
- task—Interface transactions and processing
- timer—Timer usage

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

The flags in a traceoptions flag statement are identifiers. When you use the set command to configure a flag, any flags that might already be set are not modified. In the following example, setting the timer tracing flag has no effect on the already configured task flag. Use the delete command to delete a particular flag.

```
[edit routing-options traceoptions]
user@host# show
flag task;
user@host# set traceoptions flag timer
user@host# show
flag task;
flag timer;
user@host# delete traceoptions flag task
user@host# show
flag timer;
```

Examples: Tracing Global Routing Protocol Operations

Log all globally traceable operations, saving the output in up to 10 files that are up to 10 MB in size:

```
[edit]
routing-options {
    traceoptions {
        file routing size 10m files 10;
        flag all;
    }
}
```

Log all unusual or abnormal traceable operations:

```
[edit]
routing-options {
    traceoptions {
        file routing size 10m files 10;
        flag all;
        flag normal disable;
    }
}
```

Log changes that occur in the JUNOS Software routing table:

```
[edit]
routing-options {
    traceoptions {
        file routing size 10m files 10;
        flag route;
    }
}
```

Disabling Distributed Periodic Packet Management on the Packet Forwarding Engine

Periodic packet management (PPM) is responsible for periodic transmission of packets on behalf of its various client processes, such as Bidirectional Forwarding Detection (BFD). PPM also receives packets on behalf of client processes. By default, PPM handles time-sensitive periodic processing and performs such processes as the gathering of statistics and the sending of process-specific packets. Distributing PPM to the Packet Forwarding Engine allows you to run such processes as BFD on the Packet Forwarding Engine. In JUNOS Release 9.4 and later, PPM automatically runs on both the Routing Engine and the host subsystem of the Packet Forwarding Engine or Dense Port Concentrator (DPC).

PPM runs on the Routing Engine and Packet Forwarding Engine by default. You can only disable PPM on the Packet Forwarding Engine. To disable distributed PPM on the Packet Forwarding Engine, include the ppm statement:

```
ppm {
```

```
    no-delegate-processing;
}
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

NOTE: Distributed PPM is supported only on the M7i and M10i routers with Enhanced CFEB (CFEB-E); M120 and M320 routers; and all MX Series, T Series, TX Matrix routers, and EX Series switches.

The following types of sessions are supported by distributed PPM:

- BFD single-hop session for both IPv4 and IPv6, including EBGP, ISIS, and OSPF
- Connectivity fault management (CFM) sessions
- Link fault management (LFM) sessions
- Rapid Spanning Tree Protocol (RSTP) and Multiple Spanning Tree Protocol (MSTP) interface sessions
- Link Aggregation Control Protocol (LACP) sessions (MX Series routers only)

The following types of sessions are *not* supported by distributed PPM:

- BFD over an aggregated interface for IPv4, IPv6, RSTP, MSTP, and LACP
- BFD over an IPv6 interface that does not have the global IPv6 address (or only has a link local address)
- Multihop BFD with IBGP, static routes, EBGP multihop, and MPLS LSP
- BFD over an MPLS path using OAM

In addition, on the M120 router, when Forwarding Engine Board (FEB) redundancy is configured and a FEB fails over, PPM sessions do not automatically switch over to the newly active FEB. For more information about FEB redundancy, see the *JUNOS System Basics Configuration Guide*.

Enabling Source Routing

Starting in JUNOS Release 8.2 for IPv6 and JUNOS Release 8.5 for IPv4, source routing is disabled by default on J Series Services Routers , M Series Multiservice Edge Routers, MX Series Ethernet Services Routers, T Series Core Routers, and on EX Series switches. To enable source routing, include the source-routing statement:

NOTE: We recommend that you not use source routing.

```
source-routing {
    (ip | ipv6);
}
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

Delaying Updates of the MED Path Attribute for BGP

You can configure a timer to delay update of the multiple-exit discriminator (MED) path attribute calculated for BGP groups or peers that have been configured with the metric-out igp statement. If the MED changes before the timer expires because of a change in the IGP metric associated with the route next hop, the BGP peer sends an update only if the MED is lower than the previously advertised value or another attribute associated with the route has changed, or if the BGP peer is responding to a refresh route request.

To configure an interval to delay MED IGP updates, include the med-igp-update-interval statement:

med-igp-update-interval *minutes*;

For a list of hierarchy levels at which you can include this statement, see the statement summary sections for this statement.

The default interval is 10 minutes. The interval that you can configure is in the range from 10 through 600.

You must separately configure the BGP group or peer that you want to delay sending MED IGP updates for the configured interval. For more information, see "Configuring the MED in BGP Updates" on page 746.

NOTE: If you have NSR enabled and a switchover occurs, the delayed MED updates might be advertised as soon as the switchover occurs. For more detailed information about NSR, see the *JUNOS High Availability Configuration Guide*.

Creating Policies to Control Label Allocation and Substitution for MPLS Ingress and AS Border Routers

In JUNOS Release 10.0 and later, you can control label-advertisements on MPLS ingress and AS border routers (ASBRs) on a per-route basis by specifying a label allocation policy using the allocation label-allocation-policy statement at the [edit routing-instances routing-instance-name routing-options label] hierarchy level. You can configure the label allocation mode as either per-nexthop or per-table.

Additionally, you can configure label substitution on a per-route basis by specifying a label substitution policy using the substitution label-substitution-policy statement at the [edit routing-instances routing-instance-name routing-options label] hierarchy level. The label substitution policy is used to determine whether or not a label should be substituted on an ASBR router. The results of the policy operation are either accept (label substitution is performed) or reject (label substitution is not performed). The default behavior is to accept.

For more details about allocating labels and creating substitution policies for MPLS ingress and ASBRs, see the *JUNOS VPNs Configuration Guide*.

Related Topics

- allocation statement in the *JUNOS VPNs Configuration Guide*
- substitution statement in the *JUNOS VPNs Configuration Guide*

Chapter 6

Configuring Logical Systems

This chapter discusses the following topics related to understanding and configuring logical system properties:

- Logical Systems Overview on page 135
- Logical System Configuration Statements on page 137
- Minimum Logical System Configuration on page 138
- Configuring a Logical System on page 138

Logical Systems Overview

You can partition a single physical router into multiple logical devices that perform independent routing tasks. Because logical systems perform a subset of the tasks once handled by the physical router, logical systems offer an effective way to maximize the use of a single router.

NOTE: In JUNOS Release 9.3 and later, the term *logical system* replaces *logical router*. All configuration statements, operational commands, show command outputs, error messages, log messages, and SNMP MIB objects that contain the string logical-router or logical-routers are changed to logical-system and logical-systems, respectively.

Logical systems perform a subset of the actions of a physical router and have their own unique routing tables, interfaces, policies, and routing instances. A set of logical systems within a single router can handle the functions previously performed by several small routers.

The following are supported on logical systems:

- BGP, IS-IS, LDP, OSPF, RIP, RIP next generation (RIPng), RSVP, static routes, various multicast protocols, and IP version 4 (IPv4) and version 6 (IPv6) are supported at the [edit logical-systems protocols] hierarchy level.
- Basic MPLS for core provider router functionality is supported at the [edit logical-systems protocols mpls] hierarchy level.
- All policy-related statements available at the [edit policy-options] hierarchy level are supported at the [edit logical-systems policy-options] hierarchy level.
- Most routing options statements available at the [edit routing-options] hierarchy level are supported at the [edit logical-systems routing-options] hierarchy level.

Only the route-record statement is not supported at the [edit logical-systems routing-options] hierarchy level.

- Graceful Routing Engine switchover (GRES) is supported.
- You can assign most interface types to a logical system, including SONET/SDH interfaces, Ethernet interfaces, Asynchronous Transfer Mode (ATM) interfaces, ATM2 interfaces, Channelized Q Performance Processor (QPP) interfaces, aggregated interfaces, link services interfaces, and multilink services interfaces.
- Source class usage, destination class usage, unicast reverse path forwarding, class of service, firewall filters, class-based forwarding, and policy-based accounting work with logical systems when you configure these features on the physical router.
- Multicast protocols, such as Protocol Independent Multicast (PIM) and Distance Vector Multicast Routing Protocol (DVMRP) are supported at the [edit logical-systems *logical-system-name* protocols] hierarchy level. Rendezvous point (RP) and source designated router (DR) functionality for multicast protocols within a logical system is also supported.
- The Bidirectional Forwarding Protocol (BFD) is supported.

The following restrictions apply to logical systems:

- You can configure a maximum of 15 logical systems on one physical router.
- The router has only one configuration file, which contains configuration information for the physical router and all associated logical systems. Master users can access the full configuration. However, logical system users can access only the portion of the configuration related to their particular logical system.
- All configuration commits performed by a logical system user are treated as commit private. For more information on the commit private command, see the *JUNOS System Basics Configuration Guide*.
- If a logical system experiences an interruption of its routing protocol process (rpd), the core dump output is saved in a file in the following location: /var/tmp/rpd_*logical-system-name*.core-tarball.*number*.tgz. Likewise, if you issue the restart routing command in a logical system, only the routing protocol process (rpd) for the logical system is restarted.
- If you configure trace options for a logical system, the output log file is stored in the following location: /var/tmp/*logical-system-name*.
- The following Physical Interface Cards (PICs) are not supported with logical systems: Adaptive Services PIC, ES PIC, Monitoring Services PIC, and Monitoring Services II PIC.
- Sampling, port mirroring, IP Security (IPsec), and Generalized MPLS (GMPLS) are not supported.
- Label-switched path (LSP) ping and traceroute for autonomous system (AS) number lookup are not supported.
- If you configure multiple logical systems, you can configure a VPLS routing instance only for the first logical system configured at the [edit logical-systems *logical-system-name* routing-instances *instance-name* protocols vpls] hierarchy level.

A virtual router is not the same as a logical system. A virtual router is a type of simplified routing instance that has a single routing table. A logical system is a partition of a physical router and can contain multiple routing instances and routing tables. For example, a logical system can contain multiple virtual router routing instances.

Logical System Configuration Statements

To configure logical system properties, you include statements at the [edit logical-systems *logical-system-name*] hierarchy level:

```
[edit]
logical-systems logical-system-name {
    interfaces interface-name {
        ...interface-configuration...
    }
    policy-options {
        ...policy-options-configuration...
    }
    protocols protocol {
        ...protocol-configuration...
    }
    routing-instances routing-instance-name {
        ...routing-instance-configuration...
    }
    routing-options {
        ...routing-options-configuration...
    }
}
```

As indicated, the [edit logical-systems *logical-system-name*] hierarchy level contains the following hierarchy levels in their entirety:

- [edit interfaces]
- [edit policy-options]
- [edit protocols]
- [edit routing-instances]
- [edit routing-options]

Each of these hierarchy levels is used to configure an aspect of the logical system. The logical system fully supports each subsequent hierarchy level. You always have at least one logical system, the "master" logical system by default.

For documentation of these aspects of the logical system, see the documentation for each hierarchy level. The configurations are not documented separately for logical systems.

For a detailed example of a logical system configuration, see the *JUNOS Feature Guide*.

For information on configuring logical system interface properties, see the *JUNOS Network Interfaces Configuration Guide* and the *JUNOS Services Interfaces Configuration Guide*.

For information on configuring logical system routing policy properties, see the *JUNOS Policy Framework Configuration Guide*.

For information on configuring logical system multicast protocols, see the *JUNOS Multicast Protocols Configuration Guide*.

For information on configuring logical system routing protocols, see "Interior Gateway Protocols" on page 313 and "BGP" on page 711.

For information on configuring logical system routing instances, see "Routing Instances" on page 223.

For information on configuring logical system routing options, see "Protocol-Independent Routing Properties" on page 43.

Minimum Logical System Configuration

To configure a logical system, you must include at least the following statements in the configuration:

```
[edit]
logical-systems logical-system-name {
    interfaces interface-name {
        unit unit-number {
            ...
        }
    }
}
```

Configuring a Logical System

To configure a logical system, include the logical-systems statement at the [edit] hierarchy level:

```
[edit]
logical-systems {
    logical-system-name;
}
```

Specify any logical system name to configure a logical system.

logical-systems

Syntax

```
logical-systems {
    logical-system-name {
        ...logical-system-configuration...
    }
}
```

Hierarchy Level [edit]

Release Information Statement introduced before JUNOS Release 7.4.
Statement name changed from logical-routers in JUNOS Release 9.3.

Description (M Series, MX Series, and T Series routers only) Configure a logical system.

Options *logical-system-name*—Name of the logical system.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics

- Configuring a Logical System on page 138

Chapter 7

Summary of Protocol-Independent Routing Properties Configuration Statements

This chapter provides a reference for each of the protocol-independent routing configuration statements. The statements are organized alphabetically.

active

Syntax (active | passive);

Hierarchy Level
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options (aggregate | generate | static) (defaults | route)],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options rib *routing-table-name* (aggregate | generate | static) (defaults | route)],
[edit logical-systems *logical-system-name* routing-options (aggregate | generate | static) (defaults | route)],
[edit logical-systems *logical-system-name* routing-options rib *routing-table-name* (aggregate | generate | static) (defaults | route)],
[edit routing-instances *routing-instance-name* routing-options (aggregate | generate | static) (defaults | route)],
[edit routing-instances *routing-instance-name* routing-options rib *routing-table-name* (aggregate | generate | static) (defaults | route)],
[edit routing-options (aggregate | generate | static) (defaults | route)],
[edit routing-options rib *routing-table-name* (aggregate | generate | static) (defaults | route)]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Configure whether static, aggregate, or generated routes are removed from the routing and forwarding tables when they become inactive. Routes that have been configured to remain continually installed in the routing and forwarding tables are marked with **reject** next hops when they are inactive.

- **active**—Remove a route from the routing and forwarding tables when it becomes inactive.
- **passive**—Have a route remain continually installed in the routing and forwarding tables even when it becomes inactive.

Default active

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring Static Routes on page 54
- Configuring Aggregate Routes on page 87
- Configuring Generated Routes on page 96

aggregate

Syntax

```
aggregate {
    defaults {
        ... aggregate-options ...
    }
    route destination-prefix {
        policy policy-name;
        ... aggregate-options ...
    }
}
```

Hierarchy Level

[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options rib *routing-table-name*],
[edit logical-systems *logical-system-name* routing-options],
[edit logical-systems *logical-system-name* routing-options rib *routing-table-name*],
[edit routing-instances *routing-instance-name* routing-options],
[edit routing-instances *routing-instance-name* routing-options rib *routing-table-name*],
[edit routing-options],
[edit routing-options rib *routing-table-name*]

Release Information

Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description

Configure aggregate routes.

Options

aggregate-options—Additional information about aggregate routes that is included with the route when it is installed in the routing table. Specify zero or more of the following options in *aggregate-options*. Each option is explained separately.

- (active | passive);
- as-path <*as-path*> <origin (egp | igp | incomplete)> <atomic-aggregate> <aggregator *as-number in-address*>;
- (brief | full);
- community [*community-ids*];
- discard;
- (metric | metric2 | metric3 | metric4) *value* <type *type*>;
- (preference | preference2 | color | color2) *preference* <type *type*>;
- tag *string*;

defaults—Specify global aggregate route options. These options only set default attributes inherited by all newly created aggregate routes. These are treated as global defaults and apply to all the aggregate routes you configure in the aggregate statement. This part of the aggregate statement is optional.

route *destination-prefix*—Configure a nondefault aggregate route:

- default—For the default route to the destination. This is equivalent to specifying an IP address of 0.0.0.0/0.
- *destination-prefix/prefix-length*—*destination-prefix* is the network portion of the IP address, and *prefix-length* is the destination prefix length.

The policy statement is explained separately.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics ■ Configuring Aggregate Routes on page 87

as-path

Syntax as-path <*as-path*> <aggregator *as-number ip-address*> <atomic-aggregate> <origin (egp | igp | incomplete)>;

Hierarchy Level [edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options (aggregate | generate | static) (defaults | route)],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options rib *routing-table-name* (aggregate | generate | static) (defaults | route)],
[edit logical-systems *logical-system-name* routing-options (aggregate | generate | static) (defaults | route)],
[edit logical-systems *logical-system-name* routing-options rib *routing-table-name* (aggregate | generate | static) (defaults | route)],
[edit routing-instances *routing-instance-name* routing-options (aggregate | generate | static) (defaults | route)],
[edit routing-instances *routing-instance-name* routing-options rib *routing-table-name* (aggregate | generate | static) (defaults | route)],
[edit routing-options (aggregate | generate | static) (defaults | route)],
[edit routing-options rib *routing-table-name* (aggregate | generate | static) (defaults | route)]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Associate BGP autonomous system (AS) path information with a static, aggregate, or generated route.

In JUNOS Release 9.1 and later, the numeric range for the AS number is extended to provide BGP support for 4-byte AS numbers. For more information, see "Configuring AS Numbers for BGP" on page 112. All releases of the JUNOS Software support 2-byte AS numbers.

Options aggregator—(Optional) Attach the BGP aggregator path attribute to the aggregate route. You must specify the last AS number that formed the aggregate route (encoded as two octets) for *as-number*, followed by the IP address of the BGP system that formed the aggregate route for *in-address*.

as-path—(Optional) AS path to include with the route. It can include a combination of individual AS path numbers and AS sets. Enclose sets in brackets ([ ]). The first AS number in the path represents the AS immediately adjacent to the local AS. Each subsequent number represents an AS that is progressively farther from the local AS, heading toward the origin of the path. You cannot specify a regular expression for *as-path*; you must use a full, valid AS path.

atomic-aggregate—(Optional) Attach the BGP atomic-aggregate path attribute to the aggregate route. This path attribute indicates that the local system selected a less specific route instead of a more specific route.

origin egp—(Optional) BGP origin attribute that indicates that the path information originated in another AS.

origin igp—(Optional) BGP origin attribute that indicates that the path information originated within the local AS.

origin incomplete—(Optional) BGP origin attribute that indicates that the path information was learned by some other means.

Required Privilege Level

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

Related Topics

- Configuring Static Routes on page 54
- Configuring Aggregate Routes on page 87
- Configuring Generated Routes on page 96

auto-export

Syntax

```
auto-export {
    (disable | enable);
    family {
        inet {
            multicast {
                (disable | enable);
                rib-group rib-group;
            }
            unicast {
                (disable | enable);
                rib-group rib-group;
            }
        }
        traceoptions {
            file filename <files number> <size size> <world-readable | no-world-readable>;
            flag flag <flag-modifier> <disable>;
        }
    }
}
```

Hierarchy Level

[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options],
[edit logical-systems *logical-system-name* routing-options],
[edit routing-instances *routing-instance-name* routing-options],
[edit routing-options]

Release Information

Statement introduced before JUNOS Release 7.4.

Description

Export routes between routing instances.

Options

(disable | enable)—Disable or enable auto-export.
Default: Enable

family—Address family.

inet—IP version 4 (IPv4) address family.

multicast—Multicast routing information.

unicast—Unicast routing information.

The remaining statements are explained separately.

Required Privilege Level

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

Related Topics

- Configuring Policy-Based Export for Routing Instances on page 261

autonomous-system

Syntax

```
autonomous-system autonomous-system <asdot-notation> <loops number> {
    independent-domain;
}
```

Hierarchy Level

[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options],
[edit logical-systems *logical-system-name* routing-options],
[edit routing-instances *routing-instance-name* routing-options],
[edit routing-options]

Release Information

Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.
asdot-notation introduced in JUNOS Release 9.3.
asdot-notation introduced in JUNOS Release 9.3 for EX Series switches.

Description

Specify the routing device's AS number. In JUNOS Release 9.1 and later, the numeric range is extended to provide BGP support for 4-byte AS numbers as defined in RFC 4893, *BGP Support for Four-octet AS Number Space*.

In JUNOS Release 9.3 and later, you can also configure a 4-byte AS number using the AS-dot notation format of two integer values joined by a period: < *16-bit high-order value in decimal* > . < *16-bit low-order value in decimal* > . For example, the 4-byte AS number of 65,546 in plain-number format is represented as 1.10 in the AS-dot notation format.

Options

autonomous-system—AS number. Use a number assigned to you by the Network Information Center (NIC).
Range: 1 through 4,294,967,295 ($2^{32} - 1$) in plain-number format
Range: 0.0 through 65535.65535 in AS-dot notation format

asdot-notation—(Optional) Display the configured 4-byte autonomous system number in the AS-dot notation format.
Default: Even if a 4-byte AS number is configured in the AS-dot notation format, the default is to display the AS number in the plain-number format.

number—(Optional) Maximum number of times this AS number can appear in an AS path.
Range: 1 through 10
Default: 1 (AS number can appear once)

NOTE: When you specify the same AS number in more than one routing instance on the local routing device, you must configure the same number of loops for the AS number in each instance. For example, if you configure a value of 3 for the loops statement in a VRF routing instance that uses the same AS number as that of the master instance, you must also configure a value of 3 loops for the AS number in the master instance.

Use the independent-domain option if the loops statement must be enabled only on a subset of routing instances.

The remaining statement is explained separately.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- independent-domain
- Configuring AS Numbers for BGP on page 112

bfd

Syntax

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

Hierarchy Level

[edit logical-systems *logical-system-name* protocols],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols],
[edit protocols],
[edit routing-instances *routing-instance-name* protocols]

Release Information

Statement introduced before JUNOS Release 7.4.
pipe-detail statement introduced in JUNOS Release 8.3.

Description

Configure trace options for Bidirectional Forwarding Protocol (BFD) traffic.

Default

If you do not include this statement, no BFD tracing operations are performed.

Options

disable—(Optional) Disable the BFD tracing operation. You can use this option to disable a single operation when you have defined a broad group of tracing operations, such as all.

file *filename*—Name of the file to receive the output of the tracing operation. Enclose the name in quotation marks. All files are placed in the /var/log directory . We recommend that you place global routing protocol tracing output in the routing-log file.

files *number*—(Optional) Maximum number of trace files. When a trace file named *trace-file* reaches its maximum size, it is renamed *trace-file*.0, then *trace-file*.1, and so on, until the maximum number of trace files is reached. Then the oldest trace file is overwritten.

If you specify a maximum number of files, you also must specify a maximum file size with the size option.

Range: 2 through 1000 files
Default: 2 files

flag *flag*—Tracing operation to perform. To specify more than one tracing operation, include multiple flag statements. These are the BFD protocol tracing options:

- adjacency—Trace adjacency messages.
- all—Trace all options for BFD.
- error—Trace all errors.
- event—Trace all events.
- issu—Trace in-service software upgrade (ISSU) packet activity.

- nsr-packet—Trace non-stop-routing (NSR) packet activity.
- nsr-synchronization—Trace NSR synchronization events.
- packet—Trace all packets.
- pipe—Trace pipe messages.
- pipe-detail—Trace pipe messages in detail.
- ppm-packet—Trace packet activity by periodic packet management (PPM).
- state—Trace state transitions.
- timer—Trace timer processing.

match *expression*—(Optional) Regular expression for lines to be logged.

no-world-readable—(Optional) Prevent any user from reading the log file.

size *size*—(Optional) Maximum size of each trace file, in kilobytes (KB), megabytes (MB), or gigabytes (GB). When a trace file named *trace-file* reaches this size, it is renamed *trace-file*.0. When the *trace-file* again reaches its maximum size, *trace-file*.0 is renamed *trace-file*.1 and *trace-file* is renamed *trace-file*.0. This renaming scheme continues until the maximum number of trace files is reached. Then, the oldest trace file is overwritten.

If you specify a maximum file size, you also must specify a maximum number of trace files with the files option.
Syntax: *x*k to specify KB, *x*m to specify MB, or *x*g to specify GB
Range: 10 KB through the maximum file size supported on your system
Default: 128 KB

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

Usage Guidelines See .

Required Privilege Level routing and trace—To view this statement in the configuration.
routing-control and trace–control—To add this statement to the configuration.

Related Topics ■ Tracing BFD Protocol Traffic on page 78

bfd-liveness-detection

Syntax

```
bfd-liveness-detection {
    authentication {
        algorithm algorithm-name;
        key-chain key-chain-name;
        loose-check;
    }
    detection-time {
        threshold milliseconds;
    }
    holddown-interval milliseconds;
    local-address ip-address;
    minimum-interval milliseconds;
    minimum-receive-interval milliseconds;
    minimum-receive-ttl number;
    multiplier number;
    neighbor address;
    no-adaptation;
    transmit-interval {
        threshold milliseconds;
        minimum-interval milliseconds;
    }
    version (1 | automatic);
}
```

Hierarchy Level

[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options rib *routing-table-name* static route *destination-prefix*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options rib *routing-table-name* static route *destination-prefix* qualified-next-hop (*interface-name* | *address*)],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options static route *destination-prefix*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options static route *destination-prefix* qualified-next-hop (*interface-name* | *address*)],
[edit logical-systems *logical-system-name* routing-options rib *routing-table-name* static route *destination-prefix*],
[edit logical-systems *logical-system-name* routing-options rib *routing-table-name* static route *destination-prefix* qualified-next-hop (*interface-name* | *address*)],
[edit logical-systems *logical-system-name* routing-options static route *destination-prefix*],
[edit logical-systems *logical-system-name* routing-options static route *destination-prefix* qualified-next-hop (*interface-name* | *address*)],
[edit routing-instances *routing-instance-name* routing-options rib *routing-table-name* static route *destination-prefix*],
[edit routing-instances *routing-instance-name* routing-options rib *routing-table-name* static route *destination-prefix* qualified-next-hop (*interface-name* | *address*)],
[edit routing-instances *routing-instance-name* routing-options static route *destination-prefix* qualified-next-hop (*interface-name* | *address*)],
[edit routing-instances *routing-instance-name* routing-options static route *destination-prefix*],
[edit routing-options rib *routing-table-name* static route *destination-prefix*],

[edit routing-options rib *routing-table-name* static route *destination-prefix* qualified-next-hop (*interface-name* | *address*)],
[edit routing-options static route *destination-prefix*],
[edit routing-options static route *destination-prefix* qualified-next-hop (*interface-name* | *address*)]

Release Information Statement introduced before JUNOS Release 7.4.
detection-time threshold and transmit-interval threshold options introduced in JUNOS Release 8.2.
local-address statement introduced in JUNOS Release 8.2.
minimum-receive-ttl statement introduced in JUNOS Release 8.2.
Support for logical routers introduced in JUNOS Release 8.3.
holddown-interval statement introduced in JUNOS Release 8.5.
no-adaptation statement introduced in JUNOS Release 9.0.
Support for IPv6 static routes introduced in JUNOS Release 9.1.
authentication algorithm, authentication key-chain, and authentication loose-check statements introduced in JUNOS Release 9.6.

Description Configure bidirectional failure detection timers and authentication criteria for static routes.

Options authentication algorithm *algorithm-name* —Configure the algorithm used to authenticate the specified BFD session: simple-password, keyed-md5, keyed-sha-1, meticulous-keyed-md5, or meticulous-keyed-sha-1.

authentication key-chain *key-chain-name*—Associate a security key with the specified BFD session using the name of the security keychain. The name you specify must match one of the keychains configured in the authentication-key-chains key-chain statement at the [edit security] hierarchy level.

authentication loose-check—(Optional) Configure loose authentication checking on the BFD session. Use only for transitional periods when authentication may not be configured at both ends of the BFD session.

detection-time threshold *milliseconds*—Configure a threshold. When the Bidirectional Forwarding Detection (BFD) protocol session detection time adapts to a value equal to or greater than the threshold, a single trap and a single system log message are sent.

holddown-interval *milliseconds*—Configure an interval specifying how long a BFD session must remain up before a state change notification is sent.
Range: 0 through 255,000
Default: 0

local-address *ip-address*—Enable a multihop BFD session and configure the source address for the BFD session.

minimum-interval *milliseconds*—Configure the minimum intervals at which the local routing device transmits a hello packet and then expects to receive a reply from the neighbor with which it has established a BFD session.
Range: 1 through 255,000

minimum-receive-interval *milliseconds*—Configure the minimum interval at which the local routing device expects to receive a reply from a neighbor with which it has established a BFD session.
Range: 1 through 255,000

minimum-receive-ttl *number*—Configure the time-to-live (TTL) for the multihop BFD session.
Range: 1 through 255
Default: 255

multiplier *number*—Configure number of hello packets not received by the neighbor that causes the originating interface to be declared down.
Range: 1 through 255
Default: 3

neighbor *address*—Configure a next-hop address for the BFD session for a next hop specified as an interface name.

no-adaptation—Specify for BFD sessions not to adapt to changing network conditions. We recommend that you not disable BFD adaptation unless it is preferable not to have BFD adaptation enabled in your network.

transmit-interval threshold *milliseconds*—Configure a threshold. When the BFD session transmit interval adapts to a value greater than the threshold, a single trap and a single system log message are sent. The interval threshold must be greater than the minimum transmit interval.
Range: 0 through 4,294,967,295

transmit-interval minimum-interval *milliseconds*—Configure the minimum interval at which the local routing device transmits hello packets to a neighbor with which it has established a BFD session.
Range: 1 through 255,000

version—Configure the BFD protocol version to detect.
Range: 1 or automatic
Default: automatic (autodetect the BFD protocol version)

The remaining statements are explained separately.

Required Privilege Level

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

Related Topics

- Configuring Bidirectional Forwarding Detection on page 74
- Configuring BFD Authentication for Static Routes on page 81

brief

Syntax (brief | full);

Hierarchy Level
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options (aggregate | generate) (defaults | route)],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options rib *routing-table-name* (aggregate | generate) (defaults | route)],
[edit logical-systems *logical-system-name* routing-options (aggregate | generate) (defaults | route)],
[edit logical-systems *logical-system-name* routing-options rib *routing-table-name* (aggregate | generate) (defaults | route)],
[edit routing-instances *routing-instance-name* routing-options (aggregate | generate) (defaults | route)],
[edit routing-instances *routing-instance-name* routing-options rib *routing-table-name* (aggregate | generate) (defaults | route)],
[edit routing-options (aggregate | generate) (defaults | route)],
[edit routing-options rib *routing-table-name* (aggregate | generate) (defaults | route)]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Configure all AS numbers from all contributing paths to be included in the aggregate or generated route's path.

- brief—Include only the longest common leading sequences from the contributing AS paths. If this results in AS numbers being omitted from the aggregate route, the BGP ATOMIC_ATTRIBUTE path attribute is included with the aggregate route.
- full—Include all AS numbers from all contributing paths in the aggregate or generated route's path.

Default full

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- aggregate
- generate
- Configuring Aggregate Routes on page 87
- Configuring Generated Routes on page 96

color

See preference

community

Syntax community ([*community-ids*] | no-advertise | no-export | no-export-subconfed | none);

Hierarchy Level [edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options (aggregate | generate | static) (defaults | route)],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options rib *routing-table-name* (aggregate | generate | static) (defaults | route)],
[edit logical-systems *logical-system-name* routing-options (aggregate | generate | static) (defaults | route)],
[edit logical-systems *logical-system-name* routing-options rib *routing-table-name* (aggregate | generate | static) (defaults | route)]
[edit routing-instances *routing-instance-name* routing-options (aggregate | generate | static) (defaults | route)],
[edit routing-instances *routing-instance-name* routing-options rib *routing-table-name* (aggregate | generate | static) (defaults | route)],
[edit routing-options (aggregate | generate | static) (defaults | route)],
[edit routing-options rib *routing-table-name* (aggregate | generate | static) (defaults | route)],

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Associate BGP community information with a static, aggregate, or generated route.

Options *community-ids*—One or more community identifiers. The *community-ids* format varies according to the type of attribute that you use.

The BGP community attribute format is *as-number:community-value*:

- *as-number*—AS number of the community member. It can be a value from 1 through 65,535.
- *community-value*—Identifier of the community member. It can be a number from 0 through 65,535.

For more information about BGP community attributes, see the "Configuring the Extended Communities Attribute" section in the *JUNOS Policy Framework Configuration Guide*.

For specifying the BGP community attribute only, you also can specify *community-ids* as one of the following well-known community names defined in RFC 1997:

- no-advertise—Routes containing this community name are not advertised to other BGP peers.
- no-export—Routes containing this community name are not advertised outside a BGP confederation boundary.
- no-export-subconfed—Routes containing this community name are not advertised to external BGP peers, including peers in other members' ASs inside a BGP confederation.

- none—Explicitly exclude BGP community information with a static route. Include this option when configuring an individual route in the route portion to override a community option specified in the defaults portion.

NOTE: Extended community attributes are not supported at the [edit routing-options] hierarchy level. You must configure extended communities at the [edit policy-options] hierarchy level. For information about configuring extended communities, see the *JUNOS Policy Framework Configuration Guide*.

Required Privilege Level

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

Related Topics

- aggregate
- generate
- static
- Configuring Static Routes on page 54
- Configuring Aggregate Routes on page 87
- Configuring Generated Routes on page 96

confederation

Syntax confederation *confederation-autonomous-system* members [*autonomous-systems*];

Hierarchy Level [edit logical-systems *logical-system-name* routing-options],
[edit routing-options]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Specify the routing device's confederation AS number.

Options *autonomous-system*—AS numbers of the confederation members.
Range: 1 through 65,535

confederation-autonomous-system—Confederation AS number. Use one of the numbers assigned to you by the NIC.
Range: 1 through 65,535

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring AS Confederation Members on page 113

destination-networks

Syntax destination-networks *prefix*;

Hierarchy Level [edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options dynamic-tunnels *tunnel-name*],
[edit logical-systems *logical-system-name* routing-options dynamic-tunnels *tunnel-name*],
[edit routing-instances *routing-instance-name* routing-options dynamic-tunnels *tunnel-name*],
[edit routing-options dynamic-tunnels *tunnel-name*]

Release Information Statement introduced before JUNOS Release 7.4.

Description Specify the IPv4 prefix range for the destination network by including the destination-networks statement. Only tunnels within the specified IPv4 prefix range can be created.

Options *prefix*—Destination prefix of network.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring Dynamic GRE Tunnels for VPNs on page 125

disable

Syntax disable;

Hierarchy Level [edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options graceful-restart],
[edit logical-systems *logical-system-name* routing-options graceful-restart],
[edit routing-instances *routing-instance-name* routing-options graceful-restart],
[edit routing-options graceful-restart]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Disable graceful restart.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring Graceful Restart on page 124

discard

Syntax discard;

Hierarchy Level
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options (aggregate | generate) (defaults | route)],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options rib *routing-table-name* (aggregate | generate) (defaults | route)],
[edit logical-systems *logical-system-name* routing-options (aggregate | generate) (defaults | route)],
[edit logical-systems *logical-system-name* routing-options rib *routing-table-name* (aggregate | generate) (defaults | route)],
[edit routing-instances *routing-instance-name* routing-options (aggregate | generate) (defaults | route)],
[edit routing-instances *routing-instance-name* routing-options rib *routing-table-name* (aggregate | generate) (defaults | route)],
[edit routing-options (aggregate | generate) (defaults | route)],
[edit routing-options rib *routing-table-name* (aggregate | generate) (defaults | route)]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Do not forward packets addressed to this destination. Instead, drop the packets, do not send ICMP unreachable messages to the packets' originators, and install a reject route for this destination into the routing table.

Default When an aggregate route becomes active, it is installed in the routing table with a reject next hop, which means that ICMP unreachable messages are sent.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- aggregate
- generate
- Configuring Aggregate Routes on page 87
- Configuring Generated Routes on page 96

dynamic-tunnels

Syntax
```
dynamic-tunnels tunnel-name {
    destination-networks prefix;
    source-address address;
    tunnel-type type-of-tunnel;
}
```

Hierarchy Level
```
[edit logical-systems logical-system-name routing-instances routing-instance-name
    routing-options],
[edit logical-systems logical-system-name routing-options],
[edit routing-instances routing-instance-name routing-options],
[edit routing-options]
```

Release Information Statement introduced before JUNOS Release 7.4.

Description Configure a dynamic tunnel between two PE routers.

Options *tunnel-name*—Name of the dynamic tunnel.

The remaining statements are explained separately in this chapter.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring Dynamic GRE Tunnels for VPNs on page 125

export

Syntax
```
export [ policy–names ];
```

Hierarchy Level
```
[edit logical-systems logical-system-name routing-instances routing-instance-name
    routing-options forwarding-table],
[edit logical-systems logical-system-name routing-options forwarding-table],
[edit routing-instances routing-instance-name routing-options forwarding-table],
[edit routing-options forwarding-table]
```

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Apply one or more policies to routes being exported from the routing table into the forwarding table.

Options *policy-name*—Name of one or more policies.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring Per-Packet Load Balancing on page 120
- *JUNOS Policy Framework Configuration Guide*

export-rib

Syntax export-rib *routing-table-name*;

Hierarchy Level [edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options rib-group *group-name*],
[edit logical-systems *logical-system-name* routing-options passive *group-name*],
[edit routing-instances *routing-instance-name* routing-options rib-group *group-name*],
[edit routing-options passive *group-name*]

Release Information Statement introduced before JUNOS Release 7.4.

Description Name of the routing table from which the JUNOS Software should export routing information.

Options *routing-table-name*—Routing table group name.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics

- import-rib
- passive
- Creating Routing Table Groups on page 114

fate-sharing

Syntax

```
fate-sharing {
    cost value;
    from address <to address>;
    group group-name;
}
```

Hierarchy Level

[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options],
[edit logical-systems *logical-system-name* routing-options],
[edit routing-instances *routing-instance-name* routing-options],
[edit routing-options]

Release Information

Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description

Specify a backup path in case the primary path becomes unusable.

You specify one or more objects within a group. The objects can be a LAN interface, a router ID, or a point-to-point link. Sequence is insignificant.

Changing the fate-sharing database does not affect existing established LSP until the next CSPF reoptimization. The fate-sharing database does affect fast-reroute detour path computations.

Options

group *group-name*—Each fate-sharing group must have a name, which can be up to 32 characters long and can contain letters, digits, periods (.) and hyphens (-). You can define up to 512 groups.

cost *value*—Cost assigned to the group.
Range: 1 through 65,535
Default: 1

from *address*—Address of ingress routing device.

to *address*—Address of egress routing device. For point-to-point link objects, you must specify both a from and a to address.

Required Privilege Level

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

Related Topics

- *JUNOS MPLS Applications Configuration Guide*

filter

Syntax

```
filter {
    input filter-name;
}
```

Hierarchy Level

[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options rib *routing-table-name*],
[edit logical-systems *logical-system-name* routing-options rib *routing-table-name*],
[edit routing-instances *routing-instance-name* routing-options rib *routing-table-name*],
[edit routing-options rib *routing-table-name*]

Release Information

Statement introduced before JUNOS Release 7.4.

Description

Name of the routing table from which the JUNOS Software should export routing information.

Options

input *filter-name*—Forwarding table filter name.

Required Privilege Level

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

Related Topics

- Applying Filters to the Forwarding Table on page 109

flow

Syntax

```
flow {
   route name {
      match {
         match-conditions;
      }
      term-order (legacy | standard);
      then {
         actions;
      }
   }
   validation {
      traceoptions {
         file filename <files number> <size size> <world-readable | no-world-readable>;
         flag flag <flag-modifier> <disable>;
      }
   }
}
```

Hierarchy Level [edit routing-options]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.
term-order statement introduced in JUNOS Release 10.0

Description Configure a flow route.

Default legacy

Options *actions*—An action to take if conditions match. The actions are described in "Configuring the Action for Flow Routes" on page 108.

match-conditions—Match packets to these conditions. The match conditions are described in "Configuring Match Conditions for Flow Routes" on page 106.

route *name*—Name of the flow route.

standard—Specify to use version 7 or later of the flow-specification algorithm.

term-order (legacy | standard)—Specify the version of the flow-specification algorithm.

- legacy—Use version 6 of the flow-specification algorithm.
- standard—Use version 7 of the flow-specification algorithm.

then—Actions to take on matching packets.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring Flow Routes on page 105

forwarding-cache

Syntax
```
forwarding-cache {
    threshold suppress value <reuse value>;
    timeout minutes;
}
```

Hierarchy Level
```
[edit logical-systems logical-system-name routing-instances routing-instance-name
    routing-options multicast],
[edit logical-systems logical-system-name routing-options multicast],
[edit routing-instances routing-instance-name routing-options multicast],
[edit routing-options multicast]
```

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Configure multicast forwarding cache limits.

Options The threshold statement is explained separately.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring Multicast Forwarding Cache Limits on page 119

forwarding-table

Syntax
```
forwarding-table {
    export [ policy–names ];
    (indirect-next-hop | no-indirect-next-hop);
    unicast-reverse-path (active-paths | feasible-paths);
}
```

Hierarchy Level
```
[edit logical-systems logical-system-name routing-options],
[edit routing-options]
```

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Configure information about the routing device's forwarding table.

The remaining statements are explained separately.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring Per-Packet Load Balancing on page 120

full

See brief

generate

Syntax

```
generate {
   defaults {
      generate-options;
   }
   route destination-prefix {
      policy policy-name;
      generate-options;
   }
}
```

Hierarchy Level

[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options rib *routing-table-name*],
[edit routing-options],
[edit routing-options rib *routing-table-name*]

Release Information

Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description

Configure generated routes, which are used as routes of last resort.

Options

generate-options—Additional information about generated routes, which is included with the route when it is installed in the routing table. Specify zero or more of the following options in *generate-options*. Each option is explained separately.

- (active | passive);
- as-path <*as-path*> <origin (egp | igp | incomplete)> <atomic-aggregate> <aggregator *as-number in-address*>;
- community [*community-ids*];
- discard;
- (brief | full);
- (metric | metric2 | metric3 | metric4) *value* <type *type*>;
- (preference | preference2 | color | color2) *preference* <type *type*>;
- tag *string*;

defaults—Specify global generated route options. These options only set default attributes inherited by all newly created generated routes. These are treated as global defaults and apply to all the generated routes you configure in the generate statement. This part of the generate statement is optional.

route *destination-prefix*—Configure a non-default generated route:

- default—For the default route to the destination. This is equivalent to specifying an IP address of 0.0.0.0/0.

- *destination-prefix/prefix-length*—/*destination-prefix* is the network portion of the IP address, and *prefix-length* is the destination prefix length.

The policy statement is explained separately.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring Generated Routes on page 96

graceful-restart

Syntax
```
graceful-restart {
    disable;
    restart-duration seconds;
}
```

Hierarchy Level [edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options],
[edit logical-systems *logical-system-name* routing-options],
[edit routing-instances *routing-instance-name* routing-options],
[edit routing-options]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Configure graceful restart.

The remaining statements are explained separately.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring Graceful Restart on page 124
- *JUNOS High Availability Configuration Guide*

import

Syntax import [*policy-names*];

Hierarchy Level [edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options resolution rib],
[edit logical-systems *logical-system-name* routing-options resolution rib],
[edit routing-instances *routing-instance-name* routing-options resolution rib],
[edit routing-options resolution rib]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Specify one or more import policies to use for route resolution.

Options *policy-names*—Name of one or more import policies.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring Route Resolution on page 127

import-policy

Syntax import-policy [*policy-names*];

Hierarchy Level [edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options rib-group *group-name*],
[edit logical-systems *logical-system-name* routing-options passive *group-name*],
[edit routing-instances *routing-instance-name* routing-options rib-group *group-name*],
[edit routing-options rib-groups *group-name*]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Apply one or more policies to routes imported into the routing table group. The import-policy statement complements the import-rib statement and cannot be used unless you first specify the routing tables to which routes are being imported.

Options *policy-names*—Name of one or more policies.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- export-rib
- passive
- Creating Routing Table Groups on page 114

import-rib

Syntax import-rib [*routing-table–names*];

Hierarchy Level [edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options rib-group *group-name*],
[edit logical-systems *logical-system-name* routing-options rib-group *group-name*],
[edit routing-instances *routing-instance-name* routing-options rib-group *group-name*],
[edit routing-options rib-group *group-name*]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Name of the routing table into which the JUNOS Software should import routing information. The first routing table name you enter is the primary routing table. Any additional names you enter identify secondary routing tables. When a protocol imports routes, it imports them into the primary and any secondary routing tables. If the primary route is deleted, the secondary route also is deleted. For IPv4 import routing tables, the primary routing table must be inet.0 or routing-instance-name.inet.0. For IPv6 import routing tables, the primary routing table must be inet6.0.

In JUNOS Release 9.5 and later, you can configure an IPv4 import routing table that includes both IPv4 and IPv6 routing tables. Including both types of routing tables permits you, for example, to populate an IPv6 routing table with IPv6 addresses that are compatible with IPv4. In releases prior to JUNOS Release 9.5, you could configure an import routing table with only either IPv4 or IPv6 routing tables.

Options *routing-table-names*—Name of one or more routing tables.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- export-rib
- passive
- Creating Routing Table Groups on page 114

independent-domain

Syntax independent-domain;

Hierarchy Level [edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options autonomous-system *autonomous-system*],
[edit routing-instances *routing-instance-name* routing-options autonomous-system *autonomous-system*]

Release Information Statement introduced before JUNOS Release 7.4.

Description Configure an independent AS domain.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- autonomous-system
- Configuring Independent AS Domains on page 271

indirect-next-hop

Syntax (indirect-next-hop | no-indirect-next-hop);

Hierarchy Level [edit logical-systems *logical-system-name* routing-options forwarding-table],
[edit routing-options forwarding-table]

Release Information Statement introduced in JUNOS Release 8.2.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Enable indirectly connected next hops for route convergence.

NOTE: When virtual private LAN service (VPLS) is configured on the routing device, the indirect-next-hop statement is not supported.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Enabling Indirect Next Hops on page 128

input

Syntax input *filter-name*;

Hierarchy Level [edit logical-systems *logical-system-name* routing-options rib *routing-table-name* filter],
[edit routing-options rib *routing-table-name* filter]

Release Information Statement introduced before JUNOS Release 7.4.

Description Name of the input filter.

Options *filter-name*—Name of the input filter.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Applying Filters to the Forwarding Table on page 109

install

Syntax (install | no-install);

Hierarchy Level
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options static (defaults | route)],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options rib *routing-table-name* static (defaults | route)],
[edit logical-systems *logical-system-name* routing-options rib *routing-table-name* static (defaults | route)],
[edit logical-systems *logical-system-name* routing-options static (defaults | route)],
[edit routing-instances *routing-instance-name* routing-options rib *routing-table-name* static (defaults | route)],
[edit routing-instances *routing-instance-name* routing-options static (defaults | route)],
[edit routing-options rib *routing-table-name* static (defaults | route)]
[edit routing-options static (defaults | route)]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Configure whether the JUNOS Software installs all static routes into the forwarding table. Even if you configure a route so it is not installed in the forwarding table, the route is still eligible to be exported from the routing table to other protocols.

Options install—Explicitly install all static routes into the forwarding table.

no-install—Do not install the route into the forwarding table, even if it is the route with the lowest preference.
Default: install

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- static
- Configuring Static Routes on page 54

instance-export

Syntax instance-export [*policy–names*];

Hierarchy Level [edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options],
[edit logical-systems *logical-system-name* routing-options],
[edit routing-instances *routing-instance-name* routing-options],
[edit routing-options]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Apply one or more policies to routes being exported from a routing instance.

Options *policy-names*—Name of one or more export policies.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring Policy-Based Export for Routing Instances on page 261
- *JUNOS Policy Framework Configuration Guide*

instance-import

Syntax instance-import [*policy–names*];

Hierarchy Level [edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options],
[edit logical-systems *logical-system-name* routing-options],
[edit routing-instances *routing-instance-name* routing-options],
[edit routing-options]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Apply one or more policies to routes being imported into a routing instance.

Options *policy-names*—Name of one or more import policies.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring Policy-Based Export for Routing Instances on page 261
- *JUNOS Policy Framework Configuration Guide*

interface

See the following sections:

- interface (Multicast via Static Routes) on page 180
- interface (Multicast Scoping) on page 181

interface (Multicast via Static Routes)

Syntax

```
interface interface-names {
    maximum-bandwidth bps;
    no-qos-adjust;
    reverse-oif-mapping {
        no-qos-adjust;
    }
    subscriber-leave-timer seconds;
}
```

Hierarchy Level [edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options multicast],
[edit logical-systems *logical-system-name* routing-options multicast],
[edit routing-instances *routing-instance-name* routing-options multicast],
[edit routing-options multicast]

Release Information Statement introduced in JUNOS Release 8.1.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Enable multicast traffic on an interface.

NOTE: You cannot enable multicast traffic on an interface using the enable statement and configure PIM on the same interface simultaneously.

Options *interface-name*—Name of the interface on which to enable multicast traffic. Specify the *interface-name* to enable multicast traffic on the interface.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics

- Enabling Multicast Forwarding Without PIM on page 118

interface (Multicast Scoping)

Syntax interface [*interface-names*];

Hierarchy Level [edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options multicast scope *scope-name*],
[edit logical-systems *logical-system-name* routing-options multicast scope *scope-name*],
[edit routing-instances *routing-instance-name* routing-options multicast scope *scope-name*],
[edit routing-options multicast scope *scope-name*]

Release Information Statement introduced before JUNOS Release 7.4.

Description Configure the set of interfaces for multicast scoping.

Options *interface-names*—Names of the interfaces on which to configure scoping. Specify the full interface name, including the physical and logical address components. To configure all interfaces, you can specify all. For details about specifying interfaces, see the *JUNOS Network Interfaces Configuration Guide*.

NOTE: You cannot apply a scoping policy to a specific routing instance. All scoping policies are applied to all routing instances. However, you can apply the scope statement to a specific routing instance.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- multicast
- Configuring Multicast Scoping on page 117

interface-routes

Syntax

```
interface-routes {
   family (inet | inet6) {
      export {
         lan;
         point-to-point;
      }
   }
   rib-group group-name;
}
```

Hierarchy Level

[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options],
[edit logical-systems *logical-system-name* routing-options],
[edit routing-instances *routing-instance-name* routing-options],
[edit routing-options]

Release Information

Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description

Associate a routing table group with the routing device's interfaces and specify routing table groups into which interface routes are imported.

Options

inet—Specify the IPv4 address family.

inet6—Specify the IPv6 address family.

lan—Export LAN routes.

point-to-point—Export point-to-point routes.

The remaining statement is explained separately.

Required Privilege Level

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

Related Topics

- passive
- Configuring How Interface Routes Are Imported into Routing Tables on page 116

lsp-next-hop

Syntax

```
lsp-next-hop lsp-name {
    metric metric;
    preference preference;
}
```

Hierarchy Level

[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options static route *destination-prefix*],
[edit logical-systems *logical-system-name* routing-options static route *destination-prefix*],
[edit routing-instances *routing-instance-name* routing-options static route *destination-prefix*]
[edit routing-options static route *destination-prefix*]

Release Information

Statement introduced before JUNOS Release 7.4.

Description

Specify an LSP as the next hop for a static route, and configure an independent metric or preference on that next-hop LSP.

Options

lsp-name—Name of the next-hop LSP.

metric—Metric value.
Range: 0 through 4,294,967,295 ($2^{32} - 1$)

preference—Preference value. A lower number indicates a more preferred route.
Range: 0 through 4,294,967,295 ($2^{32} - 1$)
Default: 5

Required Privilege Level

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

Related Topics

- Specifying an LSP as the Next Hop for Static Routes on page 62

martians

Syntax

```
martians {
    destination-prefix match-type <allow>;
}
```

Hierarchy Level

[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options rib *routing-table-name*],
[edit logical-systems *logical-system-name* routing-options],
[edit logical-systems *logical-system-name* routing-options rib *routing-table-name*],
[edit routing-instances *routing-instance-name* routing-options],
[edit routing-instances *routing-instance-name* routing-options rib *routing-table-name*],
[edit routing-options],
[edit routing-options rib *routing-table-name*]

Release Information

Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description

Configure martian addresses.

Options

allow—(Optional) Explicitly allow a subset of a range of addresses that has been disallowed.

destination-prefix—Destination route you are configuring:

- *destination-prefix/prefix-length*—*destination-prefix* is the network portion of the IP address, and *prefix-length* is the destination prefix length.
- default—Default route to use when routing packets do not match a network or host in the routing table. This is equivalent to specifying the IP address 0.0.0.0/0.

match-type—Criteria that the destination must match:

- exact—Exactly match the route's mask length.
- longer—The route's mask length is greater than the specified mask length.
- orlonger—The route's mask length is equal to or greater than the specified mask length.
- through *destination-prefix*—The route matches the first prefix, the route matches the second prefix for the number of bits in the route, and the number of bits in the route is less than or equal to the number of bits in the second prefix.
- upto *prefix-length*—The route's mask length falls between the two destination prefix lengths, inclusive.

Required Privilege Level

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

Related Topics ■ Configuring Martian Addresses on page 103

maximum-paths

Syntax maximum-paths *path-limit* <log-interval *seconds*> <log-only | threshold *value*>;

Hierarchy Level [edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options],
[edit logical-systems *logical-system-name* routing-options],
[edit routing-instances *routing-instance-name* routing-options],
[edit routing-options]

Release Information Statement introduced in JUNOS Release 8.0.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Configure a limit for the number of routes installed in a routing table based upon the route path.

Options log-interval *seconds*—(Optional) Minimum time interval (in seconds) between log messages.
Range: 5 through 86,400

log-only—(Optional) Sets the route limit as an advisory limit. An advisory limit triggers only a warning, and additional routes are not rejected.

path-limit—Maximum number of routes. If this limit is reached, a warning is triggered and additional routes are rejected.
Range: 1 through 4,294,967,295 ($2^{32} - 1$)
Default: No default

threshold *value*—(Optional) Percentage of the maximum number of routes that starts triggering warning. You can configure a percentage of the *path-limit* value that starts triggering the warnings.
Range: 1 through 100

NOTE: When the number or routes reaches the threshold value, routes are still installed into the routing table while warning messages are sent. When the number or routes reaches the *path-limit* value, then additional routes are rejected.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics ■ Configuring Route Limits for Routing Tables on page 271

maximum-prefixes

Syntax maximum-prefixes *prefix-limit* <log-interval *seconds*> <log-only | threshold *value*>;

Hierarchy Level [edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options],
[edit logical-systems *logical-system-name* routing-options],
[edit routing-instances *routing-instance-name* routing-options],
[edit routing-options]

Release Information Statement introduced in JUNOS Release 8.0.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Configure a limit for the number of routes installed in a routing table based upon the route prefix.

Options log-interval *seconds*—(Optional) Minimum time interval (in seconds) between log messages.
Range: 5 through 86,400

log-only—(Optional) Sets the prefix limit as an advisory limit. An advisory limit triggers only a warning, and additional routes are not rejected.

prefix-limit—Maximum number of route prefixes. If this limit is reached, a warning is triggered and any additional routes are rejected.
Range: 1 through 4,294,967,295
Default: No default

threshold *value*—(Optional) Percentage of the maximum number of prefixes that starts triggering warning. You can configure a percentage of the *prefix-limit* value that starts triggering the warnings.
Range: 1 through 100

NOTE: When the number or routes reaches the threshold value, routes are still installed into the routing table while warning messages are sent. When the number or routes reaches the *prefix-limit* value, then additional routes are rejected.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring Route Limits for Routing Tables on page 271

med-igp-update-interval

Syntax med-igp-update-interval *minutes*;

Hierarchy Level [edit routing-options]

Release Information Statement introduced in JUNOS Release 9.0
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Configure a timer for how long to delay updates for the multiple-exit discriminator (MED) path attribute for BGP groups and peers configured with the metric-out igp *offset* delay-med-update statement. The timer delays MED updates for the interval configured unless the MED is lower than the previously advertised attribute or another attribute associated with the route has changed or if the BGP peer is responding to a refresh route request.

Options *minutes*—Interval to delay MED updates.
Default: 10 minutes
Range: 10 through 600

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- metric-out
- Delaying Updates of the MED Path Attribute for BGP on page 133

metric

See the following sections:

- metric (Aggregate, Generated, or Static Route) on page 188
- metric (Qualified Next Hop on Static Route) on page 189

metric (Aggregate, Generated, or Static Route)

Syntax (metric | metric2 | metric3 | metric4) *metric* <type *type*>;

Hierarchy Level [edit logical-systems *logical-system-name* routing-options (aggregate | generate | static) (defaults | route)],
[edit routing-options (aggregate | generate | static) (defaults | route)]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Metric value for an aggregate, generated, or static route. You can specify up to four metric values, starting with metric (for the first metric value) and continuing with metric2, metric3, and metric4.

Options *metric*—Metric value.
Range: 0 through 4,294,967,295 ($2^{32} - 1$)

type *type*—(Optional) Type of route.
Range: 1 through 16

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- aggregate
- generate
- static
- Configuring a Metric Value for Static Routes on page 67
- Configuring a Metric Value for Aggregate Routes on page 90
- Configuring a Metric Value for Generated Routes on page 98

metric (Qualified Next Hop on Static Route)

Syntax metric *metric*;

Hierarchy Level [edit logical-systems *logical-system-name* routing-options static route *destination-prefix* qualified-next-hop],
[edit routing-options static route *destination-prefix* qualified-next-hop]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Metric value for a static route.

Options *metric*—Metric value.
Range: 0 through 4,294,967,295 ($2^{32} - 1$)

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- qualified-next-hop
- static
- Configuring an Independent Preference for Static Routes on page 58

multicast

Syntax

```
multicast {
    forwarding-cache {
        threshold suppress value <reuse value>;
    }
    interface interface-name {
        enable;
    }
    scope scope-name {
        interface [ interface-names ];
        prefix destination-prefix;
    }
    ssm-groups {
        address;
    }
}
```

Hierarchy Level

[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options],
[edit logical-systems *logical-system-name* routing-options],
[edit routing-instances *routing-instance-name* routing-options],
[edit routing-options]

Release Information

Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description

Configure generic multicast properties.

NOTE: You cannot apply a scoping policy to a specific routing instance. All scoping policies are applied to all routing instances. However, you can apply the scope statement to a specific routing instance.

The remaining statements are explained separately.

Required Privilege Level

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

Related Topics

- (indirect-next-hop | no-indirect-next-hop)
- Configuring Multicast Scoping on page 117
- Configuring Additional Source-Specific Multicast Groups on page 119
- *JUNOS Multicast Configuration Guide*

no-install

See install

no-readvertise

See readvertise

no-retain

See retain

nonstop-routing

Syntax nonstop-routing;

Hierarchy Level [edit logical-systems *logical-system-name* routing-options],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options],
[edit routing-instances *routing-instance-name* routing-options],
[edit routing-options]

Release Information Statement introduced in JUNOS Release 8.4.

Description For routing platforms with two Routing Engines, configure a master Routing Engine to switch over gracefully to a backup Routing Engine and to preserve routing protocol information.

Default disabled

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Enabling Nonstop Active Routing on page 129
- *JUNOS High Availability Configuration Guide*

options

Syntax

```
options {
    syslog (level level | upto level level);
}
```

Hierarchy Level

[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options],
[edit logical-systems *logical-system-name* routing-options],
[edit routing-instances *routing-instance-name* routing-options],
[edit routing-options]

Release Information

Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description

Configure the types of system logging messages sent about the routing protocols process to the system message logging file. These messages are also displayed on the system console. You can log messages at a particular level, or up to and including a particular level.

Options

level *level*—Severity of the message. It can be one or more of the following levels, in order of decreasing urgency:

- alert—Conditions that should be corrected immediately, such as a corrupted system database.
- critical—Critical conditions, such as hard drive errors.
- debug—Software debugging messages.
- emergency—Panic or other conditions that cause the system to become unusable.
- error—Standard error conditions.
- info—Informational messages.
- notice—Conditions that are not error conditions, but might warrant special handling.
- warning—System warning messages.

upto level *level*—Log all messages up to a particular level.

Required Privilege Level

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

Related Topics

- syslog in the *JUNOS System Basics Configuration Guide*
- Configuring System Logging for the Routing Protocol Process on page 126

p2mp-lsp-next-hop

Syntax
```
p2mp-lsp-next-hop {
    metric metric;
    preference preference;
}
```

Hierarchy Level
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options static route *destination-prefix*],
[edit logical-systems *logical-system-name* routing-options static route *destination-prefix*],
[edit routing-instances *routing-instance-name* routing-options static route *destination-prefix*].
[edit routing-options static route *destination-prefix*]

Release Information Statement introduced before JUNOS Release 7.4.

Description Specify a point-to-multipoint LSP as the next hop for a static route, and configure an independent metric or preference on that next-hop LSP.

Options *metric*—Metric value.
Range: 0 through 4,294,967,295 (2^{32} – 1)

preference—Preference value. A lower number indicates a more preferred route.
Range: 0 through 4,294,967,295 (2^{32} – 1)
Default: 5

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Specifying an LSP as the Next Hop for Static Routes on page 62

passive

See active

policy

Syntax policy *policy-name*;

Hierarchy Level
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options (aggregate | generate) (defaults | route)],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options rib *routing-table-name* (aggregate | generate) (defaults | route)],
[edit logical-systems *logical-system-name* routing-options (aggregate | generate) (defaults | route)],
[edit logical-systems *logical-system-name* routing-options rib *routing-table-name* (aggregate | generate) (defaults | route)],
[edit routing-instances *routing-instance-name* routing-options (aggregate | generate) (defaults | route)],
[edit routing-instances *routing-instance-name* routing-options rib *routing-table-name* (aggregate | generate) (defaults | route)],
[edit routing-options (aggregate | generate) (defaults | route)],
[edit routing-options rib *routing-table-name* (aggregate | generate) (defaults | route)]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Associate a routing policy when configuring an aggregate or generated route's destination prefix in the routes part of the aggregate or generate statement. This provides the equivalent of an import routing policy filter for the destination prefix. That is, each potential contributor to an aggregate route, along with any aggregate options, is passed through the policy filter. The policy then can accept or reject the route as a contributor to the aggregate route and, if the contributor is accepted, the policy can modify the default preferences. The contributor with the numerically smallest prefix becomes the most preferred, or *primary*, contributor. A rejected contributor still can contribute to a less specific aggregate route. If you do not specify a policy filter, all candidate routes contribute to an aggregate route.

Options *policy-name*—Name of a routing policy.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- aggregate
- generate
- Configuring Aggregate Routes on page 87
- Configuring Generated Routes on page 96

ppm

Syntax

```
ppm {
    no-delegate-processing;
}
```

Hierarchy Level [edit routing-options],
[edit logical-systems *logical-system-name* routing-options]

Release Information Statement introduced in JUNOS Release 8.2
Statement introduced in JUNOS Release 9.0 for EX Series switches.
no-delegate-processing statement introduced in JUNOS Release 9.4.
no-delegate-processing statement introduced in JUNOS Release 9.4 for EX Series switches.

Description (M120, M320, MX Series, T Series, TX Matrix routers, M7i and M10i routers with Enhanced CFEB [CFEB-E], and EX Series switches only) Disable distributed packet processing management (PPM) to the Packet Forwarding Engine.

Default enabled

Options no-delegate-processing—Disable PPM to the Packet Forwarding Engine, which is enabled by default.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics

- Disabling Distributed Periodic Packet Management on the Packet Forwarding Engine on page 131

preference

Syntax (preference | preference2 | color | color2) *preference* <type *type*>;

Hierarchy Level
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options (aggregate | generate | static) (defaults | route)],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options rib *routing-table-name* (aggregate | generate | static) (defaults | route)],
[edit logical-systems *logical-system-name* routing-options (aggregate | generate | static) (defaults | route)],
[edit logical-systems *logical-system-name* routing-options rib *routing-table-name* (aggregate | generate | static) (defaults | route)],
[edit routing-instances *routing-instance-name* routing-options (aggregate | generate | static) (defaults | route)],
[edit routing-instances *routing-instance-name* routing-options rib *routing-table-name* (aggregate | generate | static) (defaults | route)],
[edit routing-options (aggregate | generate | static) (defaults | route)],
[edit routing-options rib *routing-table-name* (aggregate | generate | static) (defaults | route)]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Preference value for a static, aggregated, or generated route. You also can specify a secondary preference value (preference2), as well as colors, which are even finer-grained preference values (color and color2).

Options *preference*—Preference value. A lower number indicates a more preferred route.
Range: 0 through 4,294,967,295 ($2^{32} - 1$)
Default: 5 (for static routes), 130 (for aggregate and generated routes)

type—(Optional) Type of route.
Range: 1 through 16

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- aggregate
- generate
- static
- Configuring Static Routes on page 54
- Configuring Aggregate Routes on page 87
- Configuring Generated Routes on page 96

prefix

Syntax prefix *destination-prefix*;

Hierarchy Level [edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options multicast scope *scope-name*],
[edit logical-systems *logical-system-name* routing-options multicast scope *scope-name*],
[edit routing-instances *routing-instance-name* routing-options multicast scope *scope-name*],
[edit routing-options multicast scope *scope-name*]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Configure the prefix for multicast scopes.

Options *destination-prefix*—Address range for the multicast scope.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- multicast
- Configuring Multicast Scoping on page 117

qualified-next-hop

Syntax

```
qualified-next-hop (address | interface-name) {
    interface interface-name;
    metric metric;
    preference preference;
}
```

Hierarchy Level

[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options static route *destination-prefix*],
[edit logical-systems *logical-system-name* routing-options rib inet6.0 static route *destination-prefix*],
[edit logical-systems *logical-system-name* routing-options static route *destination-prefix*],
[edit routing-instances *routing-instance-name* routing-options static route *destination-prefix*],
[edit routing-options rib inet6.0 static route *destination-prefix*],
[edit routing-options static route *destination-prefix*]

Release Information

Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description

Configure an independent metric or preference on a static route.

Options

address—IPv4, IPv6, or ISO network address of the next hop.

interface-name—Name of the interface on which to configure an independent metric or preference for a static route. To configure an unnumbered Ethernet interface as the next-hop interface for a static route, specify qualified-next-hop *interface-name*, where *interface-name* is the name of the IPv4 or IPv6 unnumbered Ethernet interface.

metric—Metric value.
Range: 0 through 4,294,967,295 (2^{32} – 1)

preference—Preference value. A lower number indicates a more preferred route.
Range: 0 through 4,294,967,295 (2^{32} – 1)
Default: 5

Required Privilege Level

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

Related Topics

- Configuring an Independent Preference for Static Routes on page 58

readvertise

Syntax (readvertise | no-readvertise);

Hierarchy Level
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options rib *routing-table-name* static (defaults | route)],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options static (defaults | route)],
[edit logical-systems *logical-system-name* routing-options rib *routing-table-name* static (defaults | route)],
[edit logical-systems *logical-system-name* routing-options static (defaults | route)],
[edit routing-instances *routing-instance-name* routing-options rib *routing-table-name* static (defaults | route)],
[edit routing-instances *routing-instance-name* routing-options static (defaults | route)],
[edit routing-options rib *routing-table-name* static (defaults | route)],
[edit routing-options static (defaults | route)]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Configure whether static routes are eligible to be readvertised by routing protocols:

- readvertise—Readvertise static routes.
- no-readvertise—Mark a static route as being ineligible for readvertisement; include the no-readvertise option when configuring the route.

Default readvertise

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- static
- Configuring Static Routes on page 54

resolution

Syntax

```
resolution {
    rib routing-table-name {
        import [ policy-names ];
        resolution-ribs [ routing-table-names ];
    }
}
```

Hierarchy Level

[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options],
[edit logical-systems *logical-system-name* routing-options],
[edit routing-instances *routing-instance-name* routing-options],
[edit routing-options]

Release Information

Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description

Configure route resolution.

The remaining statements are explained separately.

Required Privilege Level

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

Related Topics

- Configuring Route Resolution on page 127

resolution-ribs

Syntax

```
resolution-ribs [ routing-table-names ];
```

Hierarchy Level

[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options resolution rib],
[edit logical-systems *logical-system-name* routing-options resolution rib],
[edit routing-instances *routing-instance-name* routing-options resolution rib],
[edit routing-options resolution rib]

Release Information

Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description

Specify one or more routing tables to use for route resolution.

The remaining statements are explained separately.

Options

routing-table-names—Name of one or more routing tables.

Required Privilege Level

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

Related Topics

- Configuring Route Resolution on page 127

resolve

Syntax resolve;

Hierarchy Level [edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options rib *routing-table-name* static (defaults | route)],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options static (defaults | route)],
[edit logical-systems *logical-system-name* routing-options rib *routing-table-name* static (defaults | route)],
[edit logical-systems *logical-system-name* routing-options static (defaults | route)],
[edit routing-instances *routing-instance-name* routing-options rib *routing-table-name* static (defaults | route)],
[edit routing-instances *routing-instance-name* routing-options static (defaults | route)],
[edit routing-options rib *routing-table-name* static (defaults | route)],
[edit routing-options static (defaults | route)]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Configure statically configured routes to be resolved to a next hop that is not directly connected. The route is resolved through the inet.0 and inet.3 routing tables.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- static
- Controlling Resolution of Static Routes to Prefixes That Are Not Directly Connected on page 74

restart-duration

Syntax restart-duration *seconds*;

Hierarchy Level [edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options graceful-restart],
[edit logical-systems *logical-system-name* routing-options graceful-restart],
[edit routing-instances *routing-instance-name* routing-options graceful-restart],
[edit routing-options graceful-restart]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Configure the restart timer for graceful restart.

Options restart-duration *seconds*—Configure the time period for the restart to last.
Range: 120 through 900 seconds
Default: 90 seconds

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring Graceful Restart on page 124

retain

Syntax (retain | no-retain);

Hierarchy Level [edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options rib *routing-table-name* static (defaults | route)],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options static (defaults | route)],
[edit logical-systems *logical-system-name* routing-options rib *routing-table-name* static (defaults | route)],
[edit logical-systems *logical-system-name* routing-options static (defaults | route)],
[edit routing-instances *routing-instance-name* routing-options rib *routing-table-name* static (defaults | route)],
[edit routing-instances *routing-instance-name* routing-options static (defaults | route)],
[edit routing-options rib *routing-table-name* static (defaults | route)],
[edit routing-options static (defaults | route)]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Configure statically configured routes to be deleted from or retained in the forwarding table when the routing protocol process shuts down normally:

- retain—Have a static route remain in the forwarding table when the routing protocol process shuts down normally. Doing this greatly reduces the time required to restart a system that has a large number of routes in its routing table.
- no-retain—Delete statically configured routes from the forwarding table when the routing protocol process shuts down normally.

Default no-retain

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics

- static
- Configuring Static Routes on page 54

rib

See the following sections:

- rib (General) on page 205
- rib (Route Resolution) on page 207

rib (General)

Syntax

```
rib routing-table-name {
    aggregate {
        defaults {
            ... aggregate-options ...
        }
        route destination-prefix {
            policy policy-name;
            ... aggregate-options ...
        }
        generate {
            defaults {
                generate-options;
            }
            route destination-prefix {
                policy policy-name;
                generate-options;
            }
        }
        martians {
            destination-prefix match-type <allow>;
        }
    }
    static {
        defaults {
            static-options;
        }
        rib-group group-name;
        route destination-prefix {
            next-hop;
            static-options;
        }
    }
}
```

Hierarchy Level

[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options],
[edit logical-systems *logical-system-name* routing-options],
[edit routing-instances *routing-instance-name* routing-options],
[edit routing-options]

Release Information

Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description

Create a routing table.

Explicitly creating a routing table with the *routing-table-name* statement is optional if you are not adding any static, martian, aggregate, or generated routes to the routing table and if you also are creating a routing table group. Simply including the passive statement to indicate that a routing table is part of a routing table group is sufficient to create it.

NOTE: The IPv4 multicast routing table (inet.1) and the IPv6 multicast routing table (inet6.1) are not supported for this statement.

Default If you do not specify a routing table name with the *routing-table-name* statement, the software uses the default routing tables, which are inet.0 for unicast routes and inet.1 for the multicast cache.

Options *routing-table-name*—Name of the routing table, in the following format:
protocol [.*identifier*]

- *protocol* is the protocol family. It can be inet6 for the IPv6 family, inet for the IPv4 family, iso for the ISO protocol family, or *instance-name*.iso.0 for an ISO routing instance.
- *identifier* is a positive integer that specifies the instance of the routing table.

  Default: inet.0

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- passive
- Creating Routing Tables on page 52

rib (Route Resolution)

Syntax

```
rib routing-table-name {
    import [ policy-names ];
    resolution-ribs [ routing-table-names ];
}
```

Hierarchy Level

[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options resolution],
[edit logical-systems *logical-system-name* routing-options resolution],
[edit routing-instances *routing-instance-name* routing-options resolution],
[edit routing-options resolution]

Release Information

Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0for EX Series switches.

Description

Specify a routing table name for route resolution.

The remaining statements are explained separately.

Required Privilege Level

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

Related Topics

- Configuring Route Resolution on page 127

rib-group

Syntax rib-group *group-name*;

Hierarchy Level [edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options interface-routes],
[edit logical-systems *logical-system-name* routing-options interface-routes],
[edit logical-systems *logical-system-name* routing-options rib *routing-table-name* static],
[edit logical-systems *logical-system-name* routing-options static],
[edit routing-instances *routing-instance-name* routing-options interface-routes],
[edit routing-options interface-routes],
[edit routing-options rib *routing-table-name* static],
[edit routing-options static]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Configure which routing table groups interface routes are imported into.

Options *group-name*—Name of the routing table group. The name must start with a letter and can include letters, numbers, and hyphens. It generally does not make sense to specify more than a single routing table group.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics

- interface-routes
- rib-groups
- Configuring How Interface Routes Are Imported into Routing Tables on page 116
- Creating Routing Table Groups on page 114

rib-groups

Syntax

```
rib-groups {
    group-name {
        export-rib group-name;
        import-policy [ policy-names ];
        import-rib [ group-names ];
    }
}
```

Hierarchy Level [edit logical-systems *logical-system-name* routing-options],
[edit routing-options]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Group one or more routing tables to form a routing table group. A routing protocol can import routes into all the routing tables in the group and can export routes from a single routing table.

Each routing table group must contain one or more routing tables that the JUNOS Software uses when importing routes (specified in the import-rib statement) and optionally can contain one routing table group that the JUNOS Software uses when exporting routes to the routing protocols (specified in the export-rib statement).

Options *group-name*—Name of the routing table group. The name must start with a letter and can include letters, numbers, and hyphens.

The remaining statements are explained separately.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics

- rib-group
- Creating Routing Table Groups on page 114

route-distinguisher-id

Syntax route-distinguisher-id *address*;

Hierarchy Level [edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options],
[edit logical-systems *logical-system-name* routing-options],
[edit routing-instances *routing-instance-name* routing-options],
[edit routing-options]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Configure a route distinguisher identifier for a routing instance, specifying an IP address. If a route distinguisher is configured for a particular routing instance, that value supersedes the route distinguisher configured by this statement.

Options *address*—IP address.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring Route Distinguishers for VRF and Layer 2 VPN Instances on page 125

route-record

Syntax route-record;

Hierarchy Level [edit logical-systems *logical-system-name* routing-options],
[edit routing-options]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Export the AS path and routing information to the traffic sampling process.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring Route Recording for Flow Aggregation on page 114
- *JUNOS Network Interfaces Configuration Guide*

router-id

Syntax router-id *address*;

Hierarchy Level [edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options],
[edit logical-systems *logical-system-name* routing-options],
[edit routing-instances *routing-instance-name* routing-options],
[edit routing-options]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Specify the routing device's IP address.

NOTE: We strongly recommend that you configure the router identifier under the [edit routing-options] hierarchy level to avoid unpredictable behavior if the interface address on a loopback interface changes.

Options *address*—IP address of the routing device.
Default: Address of the first interface encountered by the JUNOS Software

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring Router Identifiers for BGP and OSPF on page 113

routing-options

Syntax routing-options { ... }

Hierarchy Level [edit],
[edit logical-systems *logical-system-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name*],
[edit routing-instances *routing-instance-name*]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Configure protocol-independent routing properties.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Protocol-Independent Routing Properties Overview on page 45

scope

Syntax

```
scope scope-name {
    interface [ interface-names ];
    prefix destination-prefix;
}
```

Hierarchy Level

[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options multicast],
[edit logical-systems *logical-system-name* routing-options multicast],
[edit routing-instances *routing-instance-name* routing-options multicast],
[edit routing-options multicast]

Release Information

Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description

Configure the set of interfaces for multicast scoping.

Options

scope-name—Name of the multicast scope.

The remaining statements are explained separately.

Required Privilege Level

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

Related Topics

- multicast
- Configuring Multicast Scoping on page 117

source-address

Syntax source-address *address*;

Hierarchy Level [edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options dynamic-tunnels *tunnel-name*,
[edit logical-systems *logical-system-name*routing-options dynamic-tunnels *tunnel-name*,
[edit routing-instances *routing-instance-name* routing-options dynamic-tunnels *tunnel-name*],
[edit routing-options dynamic-tunnels *tunnel-name*]

Release Information Statement introduced before JUNOS Release 7.4.

Description Specify the source address for the generic routing encapsulation (GRE) tunnels. The source address specifies the address used as the source for the local tunnel endpoint. This address can be any local address on the router (typically the router ID or the loopback address).

Options *address*—Name of the source address.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring Dynamic GRE Tunnels for VPNs on page 125

source-routing

Syntax
```
source-routing {
    (ip | ipv6)
}
```

Hierarchy Level [edit routing-options]

Release Information Statement for IPv6 introduced in JUNOS Release 8.2.
Statement for IPv4 introduced in JUNOS Release 8.5.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Enable source routing.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Enabling Source Routing on page 132

ssm-groups

Syntax

```
ssm-groups {
   address;
}
```

Hierarchy Level

[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options multicast]
[edit logical-systems *logical-system-name* routing-options multicast],
[edit routing-instances *routing-instance-name* routing-options multicast],
[edit routing-options multicast]

Release Information

Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description

Configure additional (source-specific multicast) SSM groups.

Options

address—Address range of the additional SSM group.

Required Privilege Level

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

Related Topics

- multicast
- Configuring Additional Source-Specific Multicast Groups on page 119

static

Syntax

```
static {
    defaults {
        static-options;
    }
    rib-group group-name;
    route destination-prefix {
        bfd-liveness-detection {
            authentication {
                algorithm algorithm-name;
                key-chain key-chain-name;
                loose-check;
            }
            detection-time {
                threshold milliseconds;
            }
            local-address ip-address;
            minimum-interval milliseconds;
            minimum-receive-interval milliseconds;
            minimum-receive-ttl number;
            multiplier number;
            neighbor address;
            no-adaptation;
            transmit-interval {
                threshold milliseconds;
                minimum-interval milliseconds;
            }
            version (1 | automatic);
        }
        next-hop address;
        next-hop options;
        qualified-next-hop address {
            metric metric;
            preference preference;
        }
        static-options;
    }
}
```

Hierarchy Level

[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options],
[edit logical-systems *logical-system-name* routing-options],
[edit logical-systems *logical-system-name* routing-options rib *routing-table-name*],
[edit routing-instances *routing-instance-name* routing-options],
[edit routing-options],
[edit routing-options rib *routing-table-name*]

Release Information

Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.
Support for BFD authentication introduced in JUNOS 9.6.
Support for BFD authentication introduced in JUNOS 9.6 for EX Series switches.

Description Configure static routes to be installed in the routing table. You can specify any number of routes within a single static statement, and you can specify any number of static options in the configuration.

Options defaults—Specify global static route options. These options only set default attributes inherited by all newly created static routes. These are treated as global defaults and apply to all the static routes you configure in the static statement. This part of the static statement is optional.

route *destination-prefix*—Destination of the static route.

- defaults—For the default route to the destination. This is equivalent to specifying an IP address of 0.0.0.0/0.
- *destination-prefix/prefix-length*—*destination-prefix* is the network portion of the IP address, and *prefix-length* is the destination prefix length.
- next-hop *address*—Reach the next-hop routing device by specifying an IP address, an interface name, or an ISO network entity title (NET).
- *nsap-prefix*—*nsap-prefix* is the network service access point (NSAP) address for ISO.

next-hop options—Additional information for how to manage forwarding of packets to the next hop.

- discard—Do not forward packets addressed to this destination. Instead, drop the packets, do not send ICMP unreachable messages to the packets' originators, and install a reject route for this destination into the routing table.
- *iso-net*—Reach the next-hop routing device by specifying an ISO NSAP.
- next-table *routing-table-name*—Name of the next routing table to the destination.
- receive—Install a receive route for this destination into the routing table.
- reject—Do not forward packets addressed to this destination. Instead, drop the packets, send ICMP unreachable messages to the packets' originators, and install a reject route for this destination into the routing table.

static-options—(Optional under route) Additional information about static routes, which is included with the route when it is installed in the routing table.

You can specify one or more of the following in *static-options*. Each of the options is explained separately.

- (active | passive);
- as-path <*as-path*> <origin (egp | igp | incomplete)> <atomic-aggregate> <aggregator *as-number in-address*>;
- community [*community-ids*];
- (install | no-install);
- (metric | metric2 | metric3 | metric4) *value* <type *type*>;
- (preference | preference2 | color | color2) *preference* <type *type*>;
- (readvertise | no-readvertise);

- (resolve | no-resolve);
- (no-retain | retain);
- tag *string*;

The remaining statements are explained separately.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring Static Routes on page 54

tag

Syntax tag *string*;

Hierarchy Level [edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options (aggregate | generate | static) (defaults | route)],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options rib *routing-table-name* (aggregate | generate | static) (defaults | route)],
[edit logical-systems *logical-system-name* routing-options (aggregate | generate | static) (defaults | route)],
[edit logical-systems *logical-system-name* routing-options rib *routing-table-name* (aggregate | generate | static) (defaults | route)],
[edit routing-instances *routing-instance-name* routing-options aggregate | generate | static) (defaults | route)],
[edit routing-instances *routing-instance-name* routing-options rib *routing-table-name* (aggregate | generate | static) (defaults | route)],
[edit routing-options (aggregate | generate | static) (defaults | route)],
[edit routing-options rib *routing-table-name* (aggregate | generate | static) (defaults | route)]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Associate an OSPF tag with a static, aggregate, or generated route.

Options *string*—OSPF tag string.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- aggregate
- generate
- static
- Configuring Static Routes on page 54
- Configuring Aggregate Routes on page 87
- Configuring Generated Routes on page 96

threshold

Syntax threshold suppress *value* <reuse *value*>;

Hierarchy Level [edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options multicast forwarding-cache],
[edit logical-systems *logical-system-name* routing-options multicast forwarding-cache],
[edit routing-instances *routing-instance-name* routing-options multicast forwarding-cache],
[edit routing-options multicast forwarding-cache]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.2 for EX Series switches.

Description Configure the suppression and reuse thresholds for multicast forwarding cache limits.

Options reuse *value*—Value at which to begin creating new multicast forwarding cache entries. This value is optional. If configured, this number should be less than the suppress value.
Range: 1 through 200,000

suppress *value*—Value at which to begin suppressing new multicast forwarding cache entries. This value is mandatory. This number should be greater than the reuse value.
Range: 1 through 200,000

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring Multicast Forwarding Cache Limits on page 119

traceoptions

Syntax

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

Hierarchy Level

[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options],
[edit logical-systems *logical-system-name* routing-options],
[edit routing-instances *routing-instance-name* routing-options],
[edit routing-options],
[edit routing-options flow]

Release Information

Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description

Define tracing operations that track all routing protocol functionality in the routing device.

To specify more than one tracing operation, include multiple flag statements.

Default

If you do not include this statement, no global tracing operations are performed.

Options

Values:

disable—(Optional) Disable the tracing operation. You can use this option to disable a single operation when you have defined a broad group of tracing operations, such as all.

file *filename*—Name of the file to receive the output of the tracing operation. Enclose the name within quotation marks. All files are placed in the directory /var/log. We recommend that you place global routing protocol tracing output in the file routing-log.

files *number*—(Optional) Maximum number of trace files. When a trace file named *trace-file* reaches its maximum size, it is renamed *trace-file*.0, then *trace-file*.1, and so on, until the maximum number of trace files is reached. Then, the oldest trace file is overwritten. Note that if you specify a maximum number of files, you also must specify a maximum file size with the size option.
Range: 2 through 1000 files
Default: 10 files

flag *flag*—Tracing operation to perform. To specify more than one tracing operation, include multiple flag statements. These are the global routing protocol tracing options:

- all—All tracing operations
- condition-manager—Condition-manager events
- config-internal—Configuration internals

- general—All normal operations and routing table changes (a combination of the normal and route trace operations)
- graceful-restart—Graceful restart operations
- normal—All normal operations
- nsr-synchronization—Nonstop active routing synchronization
- parse—Configuration parsing
- policy—Routing policy operations and actions
- regex-parse—Regular-expression parsing
- route—Routing table changes
- state—State transitions
- task—Interface transactions and processing
- timer—Timer usage

no-world-readable—(Optional) Prevent any user from reading the log file.

size *size*—(Optional) Maximum size of each trace file, in kilobytes (KB), megabytes (MB), or gigabytes (GB). When a trace file named *trace-file* reaches this size, it is renamed *trace-file*.0. When the *trace-file* again reaches its maximum size, *trace-file*.0 is renamed *trace-file*.1 and *trace-file* is renamed *trace-file*.0. This renaming scheme continues until the maximum number of trace files is reached. Then, the oldest trace file is overwritten. Note that if you specify a maximum file size, you also must specify a maximum number of trace files with the files option.
Syntax: *x*k to specify KB, *x*m to specify MB, or *x*g to specify GB
Range: 10 KB through the maximum file size supported on your system
Default: 128 KB

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

Required Privilege Level routing and trace—To view this statement in the configuration.
routing-control and trace-control—To add this statement to the configuration.

Related Topics
- Tracing Global Routing Protocol Operations on page 129

tunnel-type

Syntax tunnel-type *type*;

Hierarchy Level [edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options dynamic-tunnels *tunnel-name*],
[edit logical-systems *logical-system-name*routing-options dynamic-tunnels *tunnel-name*,
[edit routing-instances *routing-instance-name* routing-options dynamic-tunnels *tunnel-name*],
[edit routing-options dynamic-tunnels *tunnel-name*

Release Information Statement introduced before JUNOS Release 7.4.

Description Specify the type of tunnel to be dynamically created. The only valid value is gre (for GRE tunnels).

Options *type*—Tunnel type.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring Dynamic GRE Tunnels for VPNs on page 125

unicast-reverse-path

Syntax unicast-reverse-path (active-paths | feasible-paths);

Hierarchy Level [edit logical-systems *logical-system-name* routing-options forwarding-table],
[edit routing-instances *routing-instance-name* instance-type *name* routing-options forwarding-table],
[edit routing-options forwarding-table]

Release Information Statement introduced before JUNOS Release 7.4.
Support for routing instances added in JUNOS Release 8.3.

Description Control the operation of unicast reverse-path-forwarding check.

Options active-paths—Consider only active paths during the unicast reverse-path check.

feasible-paths—Consider all feasible paths during the unicast reverse-path check.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring Unicast Reverse-Path-Forwarding Check on page 122
- *JUNOS Network Interfaces Configuration Guide*

Part 3
Routing Instances

- Introduction to Routing Instances on page 225
- Routing Instances Configuration Guidelines on page 229
- Summary of Routing Instances Configuration Statements on page 273

Chapter 8

Introduction to Routing Instances

This chapter discusses the following topics:

- Routing Instances Overview on page 225

Routing Instances Overview

You can create multiple instances of BGP, IS-IS, LDP, Multicast Source Discovery Protocol (MSDP), OSPF version 2 (usually referred to simply as OSPF), OSPF version 3 (OSPFv3), Protocol Independent Multicast (PIM), RIP, and static routes by including statements at the following hierarchy levels:

- [edit routing-instances *routing-instance-name* protocols]
- [edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols]

NOTE: You can also create multiple routing instances for separating routing tables, routing policies, and interfaces for individual DHCP wholesale subscribers (retailers) in a layer 3 wholesale network. For information about how to configure layer 3 wholesale network services, see the *JUNOS Broadband Subscriber Management Solutions Guide*.

A routing instance is a collection of routing tables, interfaces, and routing protocol parameters. The set of interfaces belongs to the routing tables, and the routing protocol parameters control the information in the routing tables.

You can configure six types of routing instances: forwarding, Layer 2 virtual private network (VPN), nonforwarding, VPN routing and forwarding (VRF), virtual router, and virtual private LAN service (VPLS).

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

NOTE: The default routing instance, master, refers to the main inet.0 routing table. The master routing instance is reserved and cannot be specified as a routing instance.

Each routing instance consists of sets of the following:

- Routing tables
- Interfaces that belong to these routing tables
- Routing option configurations

You can configure eight types of routing instances:

- Forwarding—Use this routing instance type for filter-based forwarding applications. For this instance type, there is no one-to-one mapping between an interface and a routing instance. All interfaces belong to the default instance inet.0.
- Layer2-control—(MX Series routers only) Use this routing instance type for RSTP or MSTP in customer edge interfaces of a VPLS routing instance. This instance type cannot be used if the customer edge interface is multihomed to two provider edge interfaces. If the customer edge interface is multihomed to two provider edge interfaces, use the default BPDU tunneling.
- Layer 2 VPN—Use this routing instance type for Layer 2 virtual private network (VPN) implementations.
- Nonforwarding—Use this routing instance type when a separation of routing table information is required. There is no corresponding forwarding table. All routes are installed into the default forwarding table. IS-IS instances are strictly nonforwarding instance types.
- Virtual router—Similar to a VPN routing and forwarding instance type, but used for non-VPN-related applications. There are no virtual routing and forwarding (VRF) import, VRF export, VRF target, or route distinguisher requirements for this instance type.
- Virtual switch—(MX Series routers only) Use the virtual switch instance type to isolate a LAN segment with its Spanning Tree Protocol (STP) instance and separates its VLAN identifier space. For more detail information about configuring a virtual switch, see the *JUNOS MX Series Ethernet Services Routers Layer 2 Configuration Guide* and the *JUNOS MX Series Ethernet Services Routers Solutions Guide*.
- VPLS—Use the virtual private local-area network service (VPLS) routing instance type for point-to-multipoint LAN implementations between a set of sites in a VPN.
- VRF—Use the VPN routing and forwarding routing (VRF) instance type for Layer 3 VPN implementations. This routing instance type has a VPN routing table as well as a corresponding VPN forwarding table. For this instance type, there is a one-to-one mapping between an interface and a routing instance. Each VRF instance corresponds with a forwarding table. Routes on an interface go into the corresponding forwarding table.

Configure global routing options and protocols for the master instance by including statements at the [edit protocols] and [edit routing-options] hierarchy levels. Routes are installed into the master routing instance inet.0 by default, unless a routing instance is specified.

Multiple instances of BGP, OSPF, and RIP are used for Layer 3 VPN implementation. The multiple instances of BGP, OSPF, and RIP keep routing information for different VPNs separate. The VRF instance advertises routes from the customer edge (CE) router to the provider edge (PE) router and advertises routes from the PE router to the CE router. Each VPN receives only routing information belonging to that VPN.

Forwarding instances are used to implement filter-based forwarding for Common Access Layer applications.

PIM instances are used to implement multicast over VPN applications.

Nonforwarding instances of IS-IS and OSPF can be used to separate a very large network into smaller administrative entities. Instead of configuring a large number of filters, nonforwarding instances can be used to filter routes, thereby instantiating policy. Nonforwarding instances can be used to reduce the amount of routing information advertised throughout all components of a network. Routing information associated with a particular instance can be announced where required, instead of being advertised to the whole network.

Layer 2 VPN instances are used for Layer 2 VPN implementation.

Virtual router instances are similar to a VPN routing and forwarding instance type, but used for non-VPN-related applications. There are no VRF import, VRF export, VRF target, or route distinguisher requirements for this instance type.

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

For more detailed information about configuring VPNs and Layer 2 VPNs, see the *JUNOS VPNs Configuration Guide*.

For more detailed information about configuring virtual switches and Layer 2 services on MX Series routers, see the *JUNOS MX Series Ethernet Services Routers Layer 2 Configuration Guide* and the *JUNOS MX Series Ethernet Services Routers Solutions Guide*.

Chapter 9

Routing Instances Configuration Guidelines

This chapter describes the following tasks for configuring routing instances:

- Complete Routing Instances Configuration Statements on page 229
- Routing Instances Minimum Configuration on page 234
- Configuring Multiple Instances of BGP on page 240
- Configuring Multiple Instances of IS-IS on page 241
- Configuring Multiple Instances of LDP on page 245
- Configuring Multiple Instances of MSDP on page 246
- Configuring Multiple Instances of OSPF on page 247
- Configuring Multiple Instances of PIM on page 250
- Configuring Multiple Instances of RIP on page 251
- Configuring Routing Instances on page 251
- Specifying the Instance Type for Routing Instances on page 253
- Configuring Route Distinguishers for Routing Instances on page 257
- Configuring Filter-Based Forwarding on page 258
- Configuring Class-of-Service-Based Forwarding on page 259
- Configuring Secondary VRF Import and Export Policy on page 260
- Configuring Policy-Based Export for Routing Instances on page 261
- Configuring VRF Table Labels on page 265
- Configuring VRF Targets on page 265
- Configuring OSPF Domain IDs for VPNs on page 266
- Configuring Route Limits for Routing Tables on page 271
- Configuring Independent AS Domains on page 271

Complete Routing Instances Configuration Statements

To configure routing instances, include the following statements:

```
access {
... address-assignment; ...
```

```
}
access-profile profile-name;
description text;
forwarding-options;
interface interface-name;
instance-type (forwarding | layer2-control | l2vpn | no-forwarding | virtual-router |
  virtual-switch | vpls | vrf);
}
bridge-domains {
  bridge-domains-name {
    domain-type bridge;
    vlan-id (none | all | number);
    vlan-tags outer number inner number;
    interface interface-name;
    routing-interface routing-interface-name;
    bridge-options {
      mac-limit limit;
      mac-statistics;
      mac-table-size limit;
      no-mac-learning;
      static-mac mac-address;
    }
  }
}
no-vrf-advertise;
route-distinguisher (as-number:number | ip-address:number);
vrf-import [ policy-names ];
vrf-export [ policy–names ];
vrf-table-label;
vrf-target {
  export community-name;
  import community-name;
}
protocols {
  bgp {
    ... bgp-configuration ...
  }
  isis {
    ... isis-configuration ...
  }
  l2vpn {
    ... l2vpn-configuration ...
  }
  ldp {
    ... ldp-configuration ...
  }
  msdp {
    ... msdp-configuration ...
  }
  mstp {
    ... mstp-configuration ...
  }
  ospf {
    domain-id domain-id;
    domain-vpn-tag number;
    route-type-community (iana | vendor);
```

```
        ... ospf-configuration ...
    }
    ospf3 {
        domain-id domain-id;
        domain-vpn-tag number;
        route-type-community (iana | vendor);
        ... ospf3-configuration ...
    }
    pim {
        ... pim-configuration ...
    }
    rip {
        ... rip-configuration ...
    }
    ripng {
        ... ripng-configuration ...
    }
    rstp {
        ... rstp-configuration ...
    }
    vpls {
        ... vpls-configuration ...
    }
}
routing-options {
    aggregate {
        defaults {
            ... aggregate-options ...
        }
        route destination-prefix {
            policy policy-name;
            ... aggregate-options ...
        }
    }
    auto-export {
        (disable | enable);
        family {
            inet {
                multicast {
                    (disable | enable);
                    rib-group rib-group;
                }
                unicast {
                    (disable | enable);
                    rib-group rib-group;
                }
            }
        }
        traceoptions {
            file filename <files number> <size size> <world-readable | no-world-readable>;
            flag flag <flag-modifier> <disable>;
        }
    }
    autonomous-system autonomous-system <loops number> {
        independent-domain;
    }
```

```
confederation confederation-autonomous-system members autonomous-system;
fate-sharing {
   group group-name;
   cost value;
   from address [to address];
}
forwarding-table {
   export [ policy–names ];
   (indirect-next-hop | no-indirect-next-hop);
}
generate {
   defaults {
      generate-options;
   }
   route destination-prefix {
      policy policy-name;
      generate-options;
   }
}
instance-export [ policy-names ];
instance-import [ policy-names ];
interface-routes {
   rib-group group-name;
}
martians {
   destination-prefix match-type <allow>;
}
maximum-paths path-limit <log-only | threshold value log-interval seconds>;
maximum-prefixes prefix-limit <log-only | threshold value log-interval seconds>;
multicast {
   scope scope-name {
      interface [ interface-names ];
      prefix destination-prefix;
   }
   ssm-groups {
      address;
   }
}
options {
   syslog (level level | upto level);
}
rib routing-table-name {
   aggregate {
      defaults {
         ... aggregate-options ...
      }
      route destination-prefix {
         policy policy-name;
         ... aggregate-options ...
      }
   }
   generate {
      defaults {
         generate-options;
      }
      route destination-prefix {
```

```
            policy policy-name;
            generate-options;
        }
    }
    martians {
        destination-prefix match-type <allow>;
    }
    static {
        defaults {
            static-options;
        }
        rib-group group-name;
        route destination-prefix {
            lsp-next-hop{
                metric metric;
                preference preference;
            }
            next-hop;
            p2mp-lsp-next-hop {
                metric metric;
                preference preference;
            }
            qualified-next-hop address {
                interface interface-name;
                metric metric;
                preference preference;
            }
            static-options;
        }
    }
}
route-record;
router-id address;
static {
    defaults {
        static-options;
    }
    rib-group group-name;
    route destination-prefix {
        lsp-next-hop {
            metric metric;
            preference preference;
        }
        next-hop;
        p2mp-lsp-next-hop {
            metric metric;
            preference preference;
        }
        qualified-next-hop address {
            interface interface-name;
            metric metric;
            preference preference;
        }
        static-options;
    }
}
```

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

You can include the statements at the following hierarchy levels:

- [edit routing-instances *routing-instance-name*]
- [edit logical-systems *logical-system-name* routing-instances *routing-instance-name*]

Routing Instances Minimum Configuration

You can configure BGP, IS-IS, Layer 2 VPN, LDP, MSDP, OSPF, OSPFv3, PIM, RIP, RIPng, and VPLS routing instances.

This section discusses the following routing instance minimum configurations:

- Minimum Routing-Instance Configuration for BGP on page 234
- Minimum Routing-Instance Configuration for IS-IS on page 235
- Minimum Routing-Instance Configuration for Layer 2 VPNs on page 235
- Minimum Routing-Instance Configuration for LDP on page 236
- Minimum Routing-Instance Configuration for MSDP on page 236
- Minimum Routing-Instance Configuration for Multiprotocol BGP-Based Multicast VPNs on page 237
- Minimum Routing-Instance Configuration for OSPF on page 237
- Minimum Routing-Instance Configuration for OSPFv3 on page 238
- Minimum Routing-Instance Configuration for PIM on page 238
- Minimum Routing-Instance Configuration for RIP on page 239
- Minimum Routing-Instance Configuration for VPLS on page 239

Minimum Routing-Instance Configuration for BGP

To configure a routing instance for BGP, you must include at least the following statements:

```
[edit]
routing-instances {
    routing-instance-name {
        interface interface-name;
        instance-type (forwarding | l2vpn | no-forwarding | virtual-router | vpls | vrf);
        route-distinguisher (as-number:number | ip-address:number);
        vrf-import [ policy-names ];
        vrf-export [ policy-names ];
        protocols {
            bgp {
                bgp configuration;
            }
```

```
        }
    }
}
```

For more information about the BGP configuration statements, see "BGP Configuration Guidelines" on page 721. For more information about configuring VPNs, see the *JUNOS System Configuration Guide*.

Minimum Routing-Instance Configuration for IS-IS

To configure a routing instance for IS-IS, you must include at least the following statements in the configuration:

```
[edit]
routing-instances {
    routing-instance-name {
        interface interface-name;
        instance-type (forwarding | l2vpn | no-forwarding | virtual-router | vpls | vrf);
        route-distinguisher (as-number:number | ip-address:number);
        vrf-import [ policy-names ];
        vrf-export [ policy-names ];
        protocols {
            isis {
                ... isis configuration ...
            }
        }
    }
}
```

For more information about the IS-IS configuration statements, see "IS-IS Configuration Guidelines" on page 321.

Minimum Routing-Instance Configuration for Layer 2 VPNs

To create a routing instance for Layer 2 VPN, you must include at least the following statements in the configuration:

```
[edit]
routing-instances {
    routing-instance-name {
        instance-type l2vpn;
        interface interface-name;
        route-distinguisher (as-number:number | ip-address:number);
        vrf-import [ policy-names ];
        vrf-export [ policy-names ];
        protocols {
            l2vpn {
                ... l2vpn-configuration ...
            }
        }
    }
}
```

For more information about configuring Layer 2 VPNs, see the *JUNOS VPNs Configuration Guide*.

Minimum Routing-Instance Configuration for LDP

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

```
[edit]
routing-instances {
   routing-instance-name {
      instance-type (forwarding | l2vpn | no-forwarding | virtual-router | vpls | vrf);
      interface interface-name;
      route-distinguisher (as-number:number | ip-address:number);
      vrf-import [ policy-names ];
      vrf-export [ policy-names ];
      protocols {
         ldp {
            ... ldp-configuration ...
         }
      }
   }
}
```

For more information about configuring LDP, see the *JUNOS MPLS Applications Configuration Guide*.

LDP routing instances are used to support LDP over VPNs. For more information about configuring multicast over VPNs, see the *JUNOS VPNs Configuration Guide*.

Minimum Routing-Instance Configuration for MSDP

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

```
[edit]
routing-instances {
   routing-instance-name {
      instance-type (forwarding | l2vpn | no-forwarding | virtual-router | vpls | vrf);
      interface interface-name;
      route-distinguisher (as-number:number | ip-address:number);
      vrf-import [ policy-names ];
      vrf-export [ policy-names ];
      protocols {
         msdp {
            ... msdp-configuration ...
         }
      }
   }
}
```

For more information about configuring MSDP, see the *JUNOS Multicast Protocols Configuration Guide*.

Minimum Routing-Instance Configuration for Multiprotocol BGP-Based Multicast VPNs

To configure a routing instance for a multiprotocol BGP-based multicast VPN, you must include at least the following minimum configuration:

```
[edit]
routing-instances {
   routing-instance-name;
      instance-type vrf;
      interface interface-name;
      provider-tunnel {
         pim-asm {
            group-address -address;
         }
      protocols {
         mvpn;
            route-target {
               export-target {
                  target;
                  unicast;
               }
               import-target {
                  target {
                     receiver;
                     sender;
                  }
                  unicast {
                     receiver;
                     sender;
                  }
               }
            }
         }
      }
      route-distinguisher (as:number | ip-address:number);
      vrf-target community | export community-name | import community-name);
   }
}
```

For more information about Multiprotocol BGP-based Multicast VPNs, see the *JUNOS VPNs Configuration Guide* and the *JUNOS Multicast Protocols Configuration Guide*.

Minimum Routing-Instance Configuration for OSPF

To configure a routing instance for OSPF, you must include at least the following statements in the configuration:

```
[edit]
routing-instances {
   routing-instance-name {
      interface interface-name;
      instance-type (forwarding | l2vpn | no-forwarding | virtual-router | vpls | vrf);
      route-distinguisher (as-number:number | ip-address:number);
```

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

NOTE: You can configure a logical interface under only one routing instance.

For more information about the OSPF configuration statements, see "OSPF Configuration Guidelines" on page 443.

Minimum Routing-Instance Configuration for OSPFv3

To configure a routing instance for OSPFv3, you must include at least the following statements in the configuration:

```
[edit]
routing-instances {
    routing-instance-name {
        interface interface-name;
        instance-type (no-forwarding | vrf);
        vrf-import [ policy-names ];
        vrf-export [ policy-names ];
        protocols {
            ospf3 {
                ... ospf3-configuration ...
            }
        }
    }
}
```

NOTE: You can configure a logical interface under only one routing instance.

NOTE: OSPFv3 supports the no-forwarding, virtual-router, and vrf routing instance types only.

For more information about the OSPF configuration statements, see "OSPF Configuration Guidelines" on page 443.

Minimum Routing-Instance Configuration for PIM

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

```
[edit]
routing-instances {
   routing-instance-name {
      instance-type (forwarding | l2vpn | no-forwarding | virtual-router | vpls | vrf);
      interface interface-name;
      route-distinguisher (as-number:number | ip-address:number);
      vrf-import [ policy-names ];
      vrf-export [ policy-names ];
      protocols {
         pim {
            ... pim-configuration ...
         }
      }
   }
}
```

For more information about configuring PIM, see the *JUNOS Multicast Protocols Configuration Guide*.

PIM routing instances are used to support multicast over VPNs. For more detailed information about configuring multicast over VPNs, see the *JUNOS VPNs Configuration Guide*.

Minimum Routing-Instance Configuration for RIP

RIP instances are supported only for VPN routing and forwarding (VRF) instance types. This instance type provides support for Layer 3 VPNs. To configure a routing instance for RIP, you must include at least the following statements in the configuration:

```
[edit]
routing-instances {
   routing-instance-name {
      interface interface-name;
      instance-type vrf;
      route-distinguisher (as-number:number | ip-address:number);
      vrf-import [ policy-names ];
      vrf-export [ policy-names ];
      protocols {
         rip {
            ... rip-configuration ...
         }
      }
   }
}
```

For more information about the RIP configuration statements, see “RIP Configuration Guidelines” on page 583. For more information about configuring VPNs, see the *JUNOS VPNs Configuration Guide*.

Minimum Routing-Instance Configuration for VPLS

To create a routing instance for virtual private LAN services (VPLS), you must include at least the following statements in the configuration:

```
[edit]
routing-instances {
   routing-instance-name {
      instance-type vpls;
      interface interface-name;
      route-distinguisher (as-number:number | ip-address:number);
      vrf-import [ policy-names ];
      vrf-export [ policy-names ];
      protocols {
         vpls {
            ... vpls configuration ...
         }
      }
   }
}
```

For more information about configuring VPLS, see the *JUNOS VPNs Configuration Guide*. For a detailed VPLS example configuration, see the *JUNOS Feature Guide*.

Configuring Multiple Instances of BGP

You can configure multiple instances of BGP at the following hierarchy levels:

- [edit routing-instances *routing-instance-name* protocols]
- [edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols]

Multiple instances of BGP are primarily used for Layer 3 VPN support.

Currently, EBGP (nonmultihop) peers are supported under the routing-instances hierarchy. EBGP peering is established over one of the interfaces configured under the routing-instances hierarchy. Routes learned from the EBGP peer are added to the instance-name.inet.0 table by default. You can configure import and export policies to control the flow of information into and out of the instance routing table.

For Layer 3 VPN support, configure BGP on the provider edge (PE) router to receive routes from the customer edge (CE) router and to send the instances' routes to the CE router if necessary. You can use multiple instances of BGP to maintain separate per-site forwarding tables for keeping VPN traffic separate on the PE router. For more detailed information about configuring VPNs, see the *JUNOS VPNs Configuration Guide*.

You can configure import and export policies that allow the service provider to control and rate-limit traffic to and from the customer.

Example: Configuring Multiple Instances of BGP

Configure multiple instances of BGP:

```
[edit]
routing-instances {
   routing-instance-name {
```

```
        interface so-1/1/1.0;
        interface so-1/1/1.1;
        instance-type vrf;
        route distinguisher (as-number:number | ip-address:number);
        protocols {
            bgp {
                group group-name {
                    peer-as 01;
                    type external;
                    import route-name;
                    export route-name;
                    neighbor 10.0.0.1;
                }
            }
        }
    }
}
```

You can configure an EBGP multihop session for a VRF routing instance. Also, you can set up the EBGP peer between the PE and CE routers by using the loopback address of the CE router instead of the interface addresses.

Configuring Multiple Instances of IS-IS

You can configure multiple instances of IS-IS for administrative separation.

To configure multiple routing instances, perform the following tasks:

1. Configure the IS-IS default instance at the [edit protocols isis] or [edit logical-systems *logical-system-name* protocols isis] hierarchy levels with the statements needed for your network so that routes are installed in inet.0 and in the forwarding table. Make sure to include the routing table group.
2. Configure an IS-IS routing instance for each additional IS-IS routing entity, configuring the following items:
   - Interfaces
   - Routing options
   - IS-IS protocol statements belonging to that entity
   - Routing table group
3. Configure a routing table group to install routes from the routing instance into the inet.0 routing table. You can do this in two ways:
   - Create a common routing table group so that either one of two conditions is configured:
     - Routes from the routing instances are installed in inet.0 and therefore installed in the forwarding table.
     - Routes from one router in a routing instance are forwarded to another router in the same routing instance.

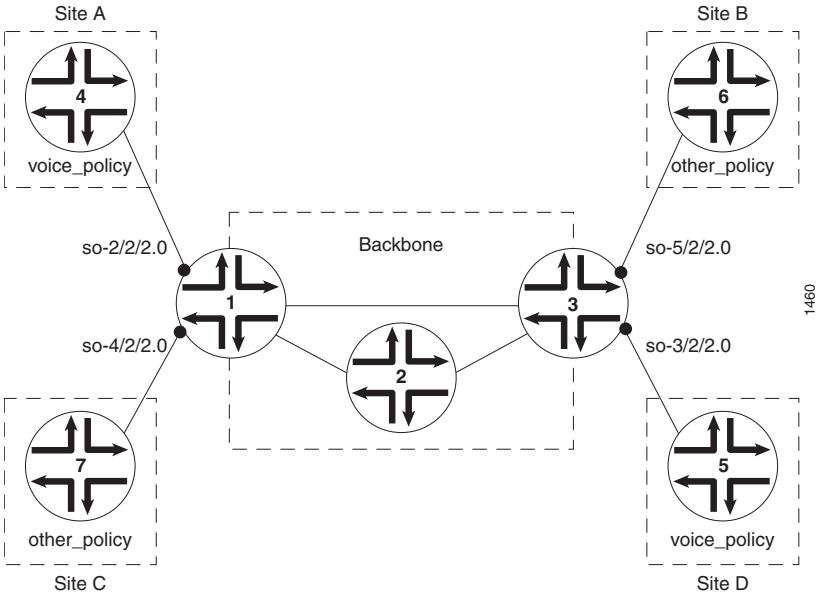
- Create a routing table group with just the routing table from one instance and inet.0 to keep the routes from going to other instances.

4. Create an export policy to export routes with a specific tag and to use that tag to export routes back into the instances. For more information, see the *JUNOS Policy Framework Configuration Guide*.

Example: Configuring Multiple Routing Instances of IS-IS

Figure 4 on page 242 shows how you can use multiple instances of IS-IS to segregate traffic within a large network. The network consists of three administrative entities: voice_policy, other_policy, and the backbone or core. Each entity is composed of several geographically separate sites that are connected by the backbone and managed by the backbone entity.

Figure 4: Configuration for Multiple Routing Instances



Sites A and D belong to the voice_policy routing instance. Sites B and C belong to the other_policy instance. Router 1 and Router 3 at the edge of the backbone connect the routing instances. Each runs a separate IS-IS instance (one per entity).

Router 1 runs three IS-IS instances: one each for Site A (voice_policy), Site C (other_policy), and the backbone, otherwise known as the default instance. Router 3 also runs three IS-IS instances: one each for Site B (other_policy), Site D (voice_policy), and the backbone (default instance).

When Router 1 runs the IS-IS instances, the following occur:

- Routes from the default instance routing table are placed in the voice_policy and other_policy instance routing tables.
- Routes from the voice_policy routing instance are placed in the default instance routing table.

- Routes from the other_policy routing instance are placed in the default instance routing table.
- Routes from the voice_policy routing instance do not enter the other_policy instance routing table.
- Routes from the other_policy routing instance do not enter the voice_policy instance routing table.

Configuring Router 1

The following sections describe how to configure Router 1 in the backbone entity with multiple routing instances.

Configure the routing instances for voice-policy and other-policy. Use all routes learned from the routing tables in the routing table group common. Export routes tagged as belonging to the routing instance.

```
[edit]
routing-instances {
   voice-policy {
      interface so-2/2/2.0;
      protocols {
         isis {
            rib-group voice_to_inet;
            export filter-on-voice-policy;
            interface so-2/2/2.0 {
               level 2 metric 20;
            }
         }
      }
   }
}
other-policy {
   interface so-4/2/2.0;
   protocols {
      isis {
         rib-group other_to_inet;
         export filter-on-other-policy;
         interface so-4/2/2.0 {
            level 2 metric 20;
         }
      }
   }
}
```

Configure the routing table group inet_to_voice_and_other to share routes with the inet.0 (in the backbone entity), voice-policy.inet.0, and other-policy.inet.0 routing tables:

```
[edit]
routing-options {
   rib-groups {
      inet_to_voice_and_other {
         import-rib [ inet.0 voice-policy.inet.0 other-policy.inet.0 ];
      }
   }
}
```

Configure the routing table group voice_to_inet to share routes with the inet.0 (in the backbone entity) and voice-policy.inet.0 routing tables:

```
[edit]
routing-options {
    rib-groups {
        voice_to_inet {
            import-rib [ voice-policy.inet.0 inet.0];
        }
    }
}
```

Configure the routing table group other_to_inet to share routes with the inet.0 (in the backbone entity) and other-policy.inet.0 routing tables:

```
[edit]
routing-options {
    rib-groups {
        other_to_inet {
            import-rib [ other-policy.inet.0 inet.0];
        }
    }
}
```

Configure the default IS-IS instance so that the routes learned from the routing instances are installed in inet.0 and the tagged routes are exported from voice-policy and other-policy:

```
[edit]
protocols {
    isis {
        export apply-tag;
        rib-group inet_to_voice_and_other;
        interface so-1/0/0.0 {
            level 2 metric 20;
        }
        interface fxp0.0 {
            disable;
        }
        interface lo0.0 {
            passive;
        }
    }
}
```

Configure routing policy for the routes learned from the routing instances:

```
[edit]
policy-options {
    policy-statement apply-tag {
        term voice-policy {
            from instance voice-policy;
            then {
                tag 10;
                accept;
            }
```

```
        }
        term other-policy {
            from instance other-policy;
            then {
                tag 12;
                accept;
            }
        }
    }
    policy-statement filter-on-voice-policy {
        from {
            tag 10;
            protocol isis;
        }
        then {
            accept;
        }
    }
    policy-statement filter-on-other-policy {
        from {
            tag 12;
            protocol isis;
        }
        then {
            accept;
        }
    }
}
```

Configuring Router 3 The configuration for Router 3 is the same as for Router 1 except that the interface names might differ. In this topology, the interface so-5/2/2.0 belongs to other-policy, and so-3/2/2.0 belongs to voice-policy.

Configuring Multiple Instances of LDP

LDP is a protocol used to distribute labels in an MPLS-enabled network.

LDP instances are used to distribute labels from a provider edge (PE) router to a customer edge (CE) router. LDP instances in a VPN are useful in carrier-of-carrier networks, where data is transmitted between two or more telecommunications carrier sites across a core provider network. Each carrier may want to restrict Internet routes strictly to the PE routers.

An advantage of using LDP instances within a VPN is that full-mesh IBGP is not required between the PE and CE routers. A router ID is required to configure an instance of LDP.

To configure multiple instances of LDP, include the following statements:

```
routing-instances {
    routing-instance-name {
        interface interface-name;
        instance-type vrf;
        protocols {
```

```
            ldp {
                ... ldp-configuration ...
            }
        }
    }
}
```

You can include the statements at the following hierarchy levels:

- [edit routing-instances *routing-instance-name* protocols]
- [edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols]

For more information about configuring LDP, see the *JUNOS MPLS Applications Configuration Guide*. For more information about configuring LDP over VPNs, see the *JUNOS VPNs Configuration Guide*.

Configuring Multiple Instances of MSDP

MSDP instances are supported only for VRF instance types. You can configure multiple instances of MSDP to support multicast over VPNs.

To configure multiple instances of MSDP, include the following statements:

```
routing-instances {
    routing-instance-name {
        interface interface-name;
        instance-type vrf;
        route-distinguisher (as-number:number | ip-address:number);
        vrf-import [ policy-names ];
        vrf-export [ policy-names ];
        protocols {
            msdp {
                ... msdp-configuration ...
            }
        }
    }
}
```

You can include the statements at the following hierarchy levels:

- [edit routing-instances *routing-instance-name* protocols]
- [edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols]

For more information about configuring MSDP, see the *JUNOS Multicast Protocols Configuration Guide*. For more information about configuring multicast over VPNs, see the *JUNOS VPNs Configuration Guide*.

Configuring Multiple Instances of OSPF

To configure multiple routing instances of OSPF or OSPFv3, perform the following tasks:

1. Configure the OSPF or OSPFv3 default instance at the [edit protocols (ospf | ospf3)] and [edit logical-systems *logical-system-name* protocols (ospf | ospf3)] hierarchy levels with the statements needed for your network so that routes are installed in inet.0 and in the forwarding table. Make sure to include the routing table group.
2. Configure an OSPF or OSPFv3 routing instance for each additional OSPF or OSPFv3 routing entity, configuring the following:
   - Interfaces
   - Routing options
   - OSPF protocol statements belonging to that entity
   - Routing table group
3. Configure a routing table group to install routes from the default route table, inet.0, into a routing instance's route table.
4. Configure a routing table group to install routes from a routing instance into the default route table, inet.0.

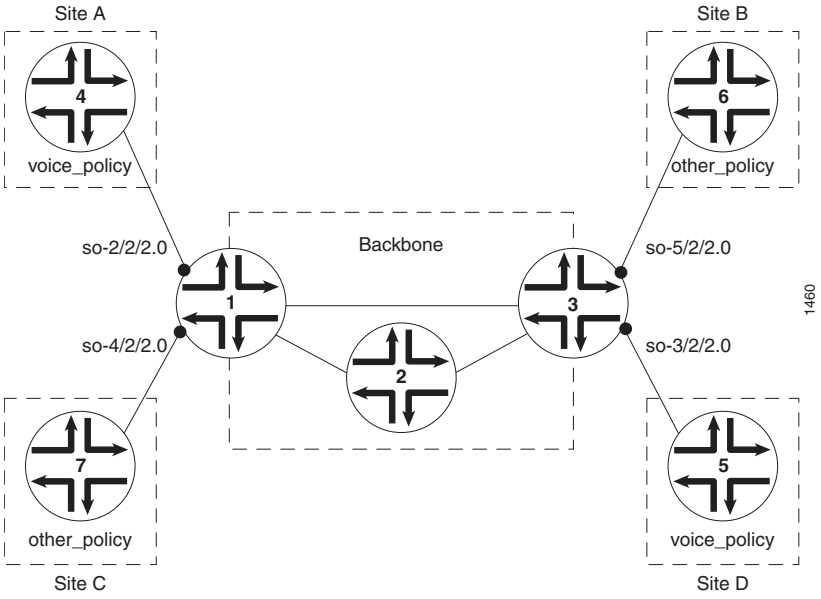
NOTE: Nonforwarding routing instances do not have forwarding tables that correspond to their routing tables.

5. Create an export policy to export routes with a specific tag and to use that tag to export routes back into the instances. For more information, see the *JUNOS Policy Framework Configuration Guide*.

Example: Configuring Multiple Routing Instances of OSPF

Figure 5 on page 248 shows how you can use multiple routing instances of OSPF or OSPFv3 to segregate prefixes within a large network. The network consists of three administrative entities: voice-policy, other-policy, and the default routing instance. Each entity is composed of several geographically separate sites that are connected by the backbone and managed by the backbone entity.

Figure 5: Configuration for Multiple Routing Instances



Sites A and D belong to the voice-policy routing instance. Sites B and C belong to the other-policy instance. Router 1 and Router 3 at the edge of the backbone connect the routing instances. Each runs a separate OSPF or OSPFv3 instance (one per entity).

Router 1 runs three OSPF or OSPFv3 instances: one each for Site A (voice-policy), Site C (other-policy), and the backbone, otherwise known as the default instance. Router 3 also runs three OSPF or OSPFv3 instances: one each for Site B (other-policy), Site D (voice-policy), and the backbone (default instance).

When Router 1 runs the OSPF or OSPFv3 instances, the following occur:

- Routes from the default instance routing table are placed in the voice_policy and other_policy instance routing tables.
- Routes from the voice-policy routing instance are placed in the default instance routing table.
- Routes from the other-policy routing instance are placed in the default instance routing table.
- Routes from the voice-policy routing instance do not enter the other-policy instance routing table.
- Routes from the other-policy routing instance do not enter the voice-policy instance routing table.

Configuring Router 1

The following sections describe how to configure Router 1 in the backbone entity with multiple routing instances.

Configure the routing instances for voice-policy and other-policy:

```
[edit]
routing-instances {
```

```
    voice-policy {
        interface so-2/2/2.0;
        protocols {
            (ospf | ospf3) {
                rib-group voice_to_inet; # Places routes into inet.0 #
                area 0.0.0.0 {
                    interface so-2/2/2.0;
                }
            }
        }
    }
    other-policy {
        interface so-4/2/2.0;
        protocols {
            (ospf | ospf3) {
                rib-group other-to-inet; # Places routes into inet.0 #
                area 0.0.0.0 {
                    interface so-4/2/2.0;
                }
            }
        }
    }
}
```

Configure the routing table group **inet-to-voice-and-others** to take routes from **inet.0** (default routing table) and place them in the **voice-policy.inet.0** and **other-policy.inet.0** routing tables:

```
[edit]
routing-options {
    rib-groups {
        inet-to-voice-and-other {
            import-rib [ inet.0 voice-policy.inet.0 other-policy.inet.0 ];
        }
    }
}
```

Configure the routing table group **voice-to-inet** to take routes from **voice-policy.inet.0** and place them in the **inet.0** default routing table:

```
[edit]
routing-options {
    rib-groups {
        voice-to-inet {
            import-rib [ voice-policy.inet.0 inet.0 ];
        }
    }
}
```

Configure the routing table group **other-to-inet** to take routes from **other-policy.inet.0** and place them in the **inet.0** default routing table:

```
[edit]
routing-options {
    rib-groups {
        other-to-inet {
```

```
            import-rib [ other-policy.inet.0 inet.0 ];
        }
    }
}
```

Configure the default OSPF or OSPFv3 instance:

```
[edit]
protocols {
    (ospf | ospf3) {
        rib-group inet-to-voice-and-other; # Place prefixes from inet.0 into
            area 0.0.0.0 { # voice-policy.inet.0 and
                interface so-2/2/2.0; # other-policy.inet.0
                interface so-4/2/2.0;
            }
        }
    }
}
```

Configuring Router 3

The configuration for Router 3 is the same as for Router 1 except that the interface names might differ. In this topology, the interface so-5/2/2.0 belongs to other-policy, and so-3/2/2.0 belongs to voice-policy.

Configuring Multiple Instances of PIM

PIM instances are supported only for VRF instance types. You can configure multiple instances of PIM to support multicast over VPNs.

To configure multiple instances of PIM, include the following statements:

```
routing-instances {
    routing-instance-name {
        interface interface-name;
        instance-type vrf;
        protocols {
            pim {
                ... pim-configuration ...
            }
        }
    }
}
```

You can include the statements at the following hierarchy levels:

- [edit routing-instances *routing-instance-name* protocols]
- [edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols]

For more information about configuring PIM, see the *JUNOS Multicast Protocols Configuration Guide*. For more information about configuring multicast over VPNs, see the *JUNOS VPNs Configuration Guide*.

Configuring Multiple Instances of RIP

RIP instances are supported only for VRF instance types. You can configure multiple instances of RIP for VPN support only. You can use RIP in the customer edge-provider edge (CE-PE) environment to learn routes from the CE router and to propagate the PE router's instance routes in the CE router.

RIP routes learned from neighbors configured under any instance hierarchy are added to the instance's routing table, *instance-name*.inet.0.

RIP does not support routing table groups; therefore, it cannot import routes into multiple tables as the OSPF or OSPFv3 protocol does.

To configure multiple instances of RIP, include the following statements:

```
routing-instances {
    routing-instance-name {
        interface interface-name;
        instance-type vrf;
        protocols {
            rip {
                interface interface-name;
                neighbor ip-address;
            }
        }
    }
}
```

You can include the statements at the following hierarchy levels:

- [edit routing-instances *routing-instance-name* protocols]
- [edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols]

Configuring Routing Instances

You can create multiple instances of BGP, IS-IS, OSPF, OSPFv3, RIP, and static routes. For information about how to configure a virtual switch, see the *JUNOS MX Series Ethernet Services Routers Layer 2 Configuration Guide*.

You can include the statements at the following hierarchy levels:

- [edit routing-instances *routing-instance-name* protocols]
- [edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols]

Each routing instance consist of the following:

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

- A set of routing option configurations

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

Configure global routing options and protocols for the default instance by including statements.

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for these statements.

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

NOTE: In JUNOS Release 9.0 and later, you can no longer specify a routing-instance name of *default* or include special characters within the name of a routing instance.

For details about specifying interfaces, see the *JUNOS Network Interfaces Configuration Guide*.

To configure a routing instance, include the following statements:

```
routing-instances {
    routing-instance-name {
        interface interface-name;
        instance-type (forwarding | layer2-control | l2vpn | no-forwarding | virtual-router |
            virtual-switch | vpls | vrf);
        no-vrf-advertise;
        route-distinguisher (as-number:number | ip-address:number);
        vrf-import [ policy-names ];
        vrf-export [ policy-names ];
        vrf-table-label;
        protocols {
            bgp {
                ... bgp-configuration ...
            }
            isis {
                isis-configuration;
            }
            l2vpn {
                l2vpn-configuration;
            }
            ldp {
                ... ldp-configuration ...
            }
            msdp {
                msdp-configuration;
            }
            ospf {
                domain-id domain-id;
```

```
            domain-vpn-tag number;
            route-type-community (iana | vendor);
            ospf-configuration;
        }
        ospf3 {
            domain-id domain-id;
            domain-vpn-tag number;
            route-type-community (iana | vendor);
            ospf3-configuration;
        }
        pim {
            pim-configuration;
        }
        rip {
            rip-configuration;
        }
        ripng {
            ripng-configuration;
        }
        vpls {
            vpls-configuration;
        }
    }
  }
}
```

Specifying the Instance Type for Routing Instances

You can configure eight routing instance types at the [edit routing-instances *routing-instance-name* instance-type] and [edit logical-systems *logical-system-name* routing-instances *routing-instance-name* instance-type] hierarchy levels:

- Forwarding—Use this routing instance type for filter-based forwarding applications. For this instance type, there is no one-to-one mapping between an interface and a routing instance. All interfaces belong to the default instance inet.0.
- Layer 2 VPN—Use this routing instance type for Layer 2 VPN implementations.
- Layer 2-control—(MX Series routers only) Use this routing instance type for RSTP or MSTP in customer edge interfaces of a VPLS routing instance. This instance type cannot be used if the customer edge interface is multihomed to two provider edge interfaces. If the customer edge interface is multihomed to two provider edge interfaces, use the default BPDU tunneling. For more information about configuring a layer2-control instance type, see the *JUNOS MX Series Ethernet Services Routers Layer 2 Configuration Guide*.
- No-forwarding—Use this routing instance type when a separation of routing table information is required. There is no corresponding forwarding table. All routes are installed into the default forwarding table. IS-IS instances are strictly nonforwarding instance types.
- Virtual router—This routing instance is similar to a VPN routing and forwarding instance type, but used for non-VPN-related applications. There are no VRF import, VRF export, VRF target, or route distinguisher requirements for this instance type.

- Virtual switch—(MX Series routers only) Use the virtual switch instance type to isolate a LAN segment with its Spanning Tree Protocol (STP) instance and separates its VLAN identifier space. For more information about configuring a virtual switch instance type, see the *JUNOS MX Series Ethernet Services Routers Layer 2 Configuration Guide*. and the *JUNOS MX Series Ethernet Services Routers Solutions Guide*.
- VPLS—Use this routing instance type for point-to-multipoint LAN implementations between a set of sites in a VPN.
- VRF—Use this routing instance type for Layer 3 VPN implementations. For this instance type, there is a one-to-one mapping between an interface and a routing instance. Each VRF instance corresponds with a forwarding table. Routes on an interface go into the corresponding forwarding table.

To configure a routing instance type, include the instance-type statement:

```
routing-instances {
    routing-instance-name {
        interface interface-name;
        instance-type (forwarding | l2vpn | layer2-control | no-forwarding | virtual-router |
            virtual-switch | vpls | vrf);
    }
}
```

You can include the statement at the following hierarchy levels:

- [edit routing-instances *routing-instance-name*]
- [edit logical-systems *logical-system-name* routing-instances *routing-instance-name*]

For more information about configuring Layer 2 VPNs, Layer 3 VPNs, and VPLS, see the *JUNOS VPNs Configuration Guide*.

For more information about configuring the types of routing instances, see the following sections:

- Configuring VRF Routing Instances on page 254
- Configuring Non-VPN VRF Routing Instances on page 255
- Configuring VPLS Routing Instances on page 256

Configuring VRF Routing Instances

To configure a VPN VRF routing instance, include the following statements:

```
interface interface-name;
instance-type vrf;
no-vrf-advertise;
route-distinguisher (as-number:number | ip-address:number);
vrf-import [ policy-names ];
vrf-export [ policy-names ];
vrf-table-label;
protocols {
    bgp {
```

```
        ... bgp-configuration ...
    }
    isis {
        ... isis-configuration ...
    }
    l2vpn {
        ... l2vpn-configuration ...
    }
    ldp {
        ... ldp-configuration ...
    }
    msdp {
        ... msdp-configuration ...
    }
    ospf {
        domain-id domain-id;
        domain-vpn-tag number;
        route-type-community (iana | vendor);
        ... ospf-configuration ...
    }
    ospf3 {
        domain-id domain-id;
        domain-vpn-tag number;
        route-type-community (iana | vendor);
        ... ospf3-configuration ...
    }
    pim {
        ... pim-configuration ...
    }
    rip {
        ... rip-configuration ...
    }
    vpls {
        ... vpls-configuration ...
    }
}
```

You can include the statements at the following hierarchy levels:

- [edit routing-instances *routing-instance-name*]
- [edit logical-systems *logical-system-name* routing-instances *routing-instance-name*]

Configuring Non-VPN VRF Routing Instances

To configure a non-VPN VRF routing instance (for example, to allow IPsec tunnels within VRF routing instances), include the following statements:

```
interface interface-name;
instance-type virtual-router;
protocols {
    bgp {
        ... bgp-configuration ...
    }
    isis {
```

```
        ,,, isis-configuration ...
    }
    ldp {
        ... ldp-configuration ...
    }
    msdp {
        ... msdp-configuration ...
    }
    ospf {
        domain-id domain-id;
        domain-vpn-tag number;
        route-type-community (iana | vendor);
        ... ospf-configuration ...
    }
    ospf3 {
        domain-id domain-id;
        domain-vpn-tag number;
        route-type-community (iana | vendor);
        ... ospf3-configuration ...
    }
    pim {
        ... pim-configuration ...
    }
    rip {
        ... rip-configuration ...
    }
}
```

You can include the statements at the following hierarchy levels:

- [edit routing-instances *routing-instance-name*]
- [edit logical-systems *logical-system-name* routing-instances *routing-instance-name*]

Configuring VPLS Routing Instances

To configure a VPLS routing instance, include the following statements:

```
interface interface-name;
instance-type vpls;
protocols {
    vpls {
        ... vpls-configuration ...
    }
}
```

You can include the statements at the following hierarchy levels:

- [edit routing-instances *routing-instance-name*]
- [edit logical-systems *logical-system-name* routing-instances *routing-instance-name*]

For more detailed information about configuring VPLS and Layer 2 VPN, see the *JUNOS VPNs Configuration Guide* and the *JUNOS Feature Guide*.

Configuring Route Distinguishers for Routing Instances

Each routing instance must have a unique route distinguisher associated with it. The route distinguisher is used to place bounds around a VPN so the same IP address prefixes can be used in different VPNs without having them overlap.

We recommend that you use a unique route distinguisher for each routing instance that you configure. Although you could use the same route distinguisher on all PE routers for the same VPN, if you use a unique route distinguisher, you can determine the CE router from which a route originated.

To configure a route distinguisher, include the route-distinguisher statement:

```
route-distinguisher (as-number:number | ip-address:number);
}
```

You can include the statement at the following hierarchy levels:

- [edit routing-instances *routing-instance-name*]
- [edit logical-systems *logical-system-name* routing-instances *routing-instance-name*]

The route distinguisher is a 6-byte value that you can specify in one of the following formats:

- *as-number*:*number*, where *as-number* is your assigned AS number and *number* is any 2-byte or 4-byte value. The AS number can be in the range from 1 through 4,294,967,295. If the AS number is a 2-byte value, the administrative number is a 4-byte value. If the AS number is 4-byte value, the administrative number is a 2-byte value.

NOTE: In JUNOS Release 9.1 and later, the numeric range for AS numbers is extended to provide BGP support for 4-byte AS numbers, as defined in RFC 4893, *BGP support for Four-octet AS Number Space*. All releases of the JUNOS Software support 2-byte AS numbers.

In JUNOS Release 9.2 and later, you can also configure a 4-byte AS number using the AS-dot notation format of two integer values joined by a period: < *16-bit high-order value in decimal* > . < *16-bit low-order value in decimal* > . For example, the 4-byte AS number of 65,546 in plain-number format is represented as 1.10 in the AS-dot notation format.

- *ip-address*:*number*, where *ip-address* is an IP address in your assigned prefix range (a 4-byte value) and *number* is any 2-byte value. The IP address can be in the range from 0 through 4,294,967,295 ($2^{32} - 1$).

Configuring Filter-Based Forwarding

You can create a filter to classify packets to determine their forwarding path within a router. Use filter-based forwarding to redirect traffic for analysis.

Filter-based forwarding is supported for IP version 4 (IPv4) and IP version 6 (IPv6).

Use filter-based forwarding for service provider selection when customers have Internet connectivity provided by different ISPs yet share a common access layer. When a shared media (such as a cable modem) is used, a mechanism on the common access layer looks at Layer 2 or Layer 3 addresses and distinguishes between customers. You can use filter-based forwarding when the common access layer is implemented using a combination of Layer 2 switches and a single router.

With filter-based forwarding, all packets received on an interface are considered. Each packet passes through a filter that has match conditions. If the match conditions are met for a filter and you have created a routing instance, filter-based forwarding is applied to a packet. The packet is forwarded based on the next hop specified in the routing instance. For static routes, the next hop can be a specific LSP. For more information about configuring LSPs, see the *JUNOS MPLS Applications Configuration Guide*.

NOTE: Source-class usage filter matching and unicast reverse-path forwarding checks are not supported on an interface configured with filter-based forwarding (FBF).

To configure filter-based forwarding, perform the following tasks:

- Create a match filter on an ingress router. To specify a match filter, include the filter *filter-name* statement at the [edit firewall] hierarchy level. For more information about creating a match filter for packet forwarding, see the *JUNOS Policy Framework Configuration Guide*. A packet that passes through the filter is compared against a set of rules to classify it and to determine its membership in a set. Once classified, the packet is forwarded to a routing table specified in the accept action in the filter description language. The routing table then forwards the packet to the next hop that corresponds to the destination address entry in the table.
- Create routing instances that specify the routing table(s) to which a packet is forwarded, and the destination to which the packet is forwarded at the [edit routing-instances] or [edit logical-systems *logical-system-name* routing-instances] hierarchy levels. For example:

```
[edit]
routing-instances {
   routing-table-name1 {
      instance-type forwarding;
      routing-options {
         static {
            route 0.0.0.0/0 nexthop 10.0.0.1;
         }
      }
```

```
    }
    routing-table-name2 {
        instance-type forwarding;
        routing-options {
            static {
                route 0.0.0.0/0 nexthop 10.0.0.2;
            }
        }
    }
}
```

- Create a routing table group that adds interface routes to the forwarding routing instances used in filter-based forwarding (FBF), as well as to the default routing instance inet.0. This part of the configuration resolves the routes installed in the routing instances to directly connected next hops on that interface. Create the routing table group at the [edit routing-options] or [edit logical-systems *logical-system-name* routing-options] hierarchy levels.

For IPv4, the following configuration installs interface routes into the default routing instance inet.0, as well as two forwarding routing instances—routing-table-name1.inet.0 and routing-table-name2.inet.0:

```
[edit]
routing-options {
    interface-routes {
        rib-group inet group-name;
    }
    rib-groups {
        group-name {
            import-rib [ inet.0 routing-table-name1.inet.0
            routing-table-name2.inet.0 ];
        }
    }
}
```

NOTE: Specify inet.0 as one of the routing instances that the interface routes are imported into. If the default instance inet.0 is not specified, interface routes are not imported into the default routing instance.

Configuring Class-of-Service-Based Forwarding

Class-of-service (CoS)-based forwarding allows you to control the next-hop selection based on a packet's class of service or IP precedence. It allows path selection based on a multifield classifier.

To configure CoS-based forwarding, perform the following tasks:

1. Create a routing policy at the [edit policy-options] or [edit logical-systems *logical-system-name* policy-options] hierarchy levels to limit the configuration so that routes matching the route filter are subject to the CoS next-hop mapping specified in my-cos-map:

```
[edit]
policy-options {
   policy-statement my-cos-forwarding {
      from {
         route-filter ...;
      }
      then {
         cos-next-hop-map my-cos-map;
      }
   }
}
```

2. Create a CoS next-hop map. To specify a CoS next-hop map, include the cos-next-hop-map statement at the [edit class-of-service] hierarchy level. For more information about creating a CoS next-hop map, see the *JUNOS Class of Service Configuration Guide*.

3. Specify the exporting of the routes to the forwarding table at the [edit routing-options] or [edit logical-systems *logical-system-name* routing-options] hierarchy levels:

```
[edit]
routing-options {
   forwarding-table {
      export my-cos-forwarding;
   }
}
```

4. Specify a static route that has multiple next hops for load balancing at the [edit routing-options] or [edit logical-systems *logical-system-name* routing-options] hierarchy levels:

```
[edit]
routing-options {
   static {
      route 12.1.1.1/32 {
         next-hop [ 3.1.1.2 3.1.1.4 3.1.1.6 3.1.1.8 ];
      }
   }
}
```

Configuring Secondary VRF Import and Export Policy

You configure a VPN routing and forwarding instance (VRF) so that routes received from the provider edge-provider edge (PE-PE) session (in the default instance) can be imported into any of an instance's VRF secondary routing tables. Importing depends on defined policies. Routes to be exported should pass through the policies listed in the export list.

To configure secondary VRF import and export policies, include the following statements:

```
[edit]
routing-instances {
```

```
    routing-instance-name {
        instance-type vrf;
        vrf-import [ policy-names ];
        vrf-export [ policy-names ];
    }
}
policy-options {
    policy-statement policy-name {
        from community community-name;
        then accept;
    }
}
```

For more information about configuring VPNs, see the *JUNOS VPNs Configuration Guide*.

Configuring Policy-Based Export for Routing Instances

Configuring policy-based export simplifies the process of exchanging route information between routing instances.

Exporting routing information between routing instances typically is accomplished by configuring separate routing table groups for each instance. The use of policy-based export reduces the configuration needed for exporting routes between multiple routing instances by eliminating the configuration of separate routing table groups for each instance.

Policy-based export is particularly useful in the following two cases:

- Overlapping VPNs—VPN configurations in which more than one VRF has the same route target
- Nonforwarding instances—Multilevel IGPs using multiple routing instances

NOTE: The instance-export and instance-import statements are not valid for VRF instances. The auto-export statement is valid for VRF and non-VRF instances. The instance-import statement automatically enables auto-export for non-VRF instances.

For detailed information about configuring overlapping VPNs and nonforwarding instances, see the *JUNOS VPNs Configuration Guide*.

For sample configurations, see the following sections:

- Example: Configuring Policy-Based Export for an Overlapping VPN on page 261
- Example: Configuring Policy-Based Export for a Nonforwarding Instance on page 263

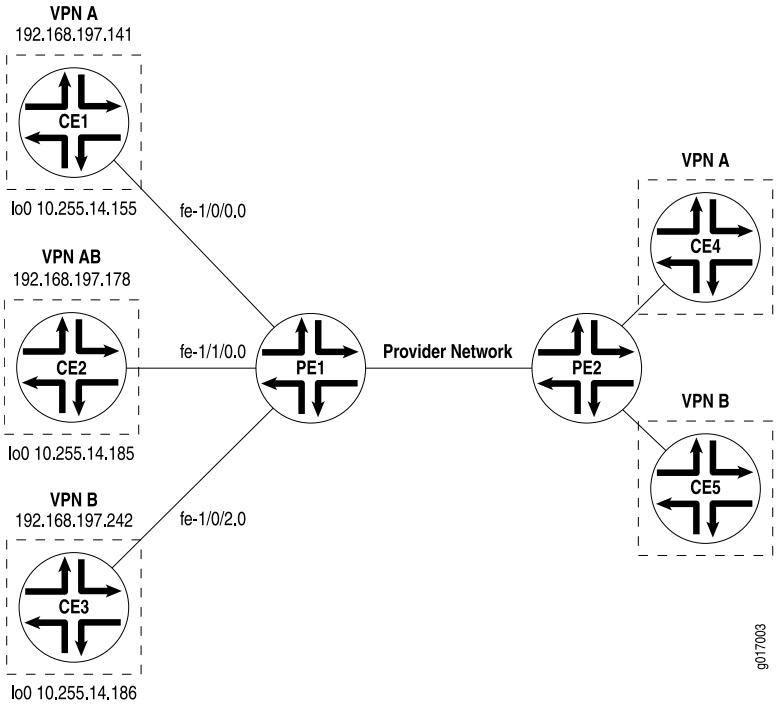
Example: Configuring Policy-Based Export for an Overlapping VPN

In Layer 3 VPNs, a CE router is often a member of more than one VPN. Figure 6 on page 262 illustrates the topology for the configuration example in this section. The

configurations in this section illustrate local connectivity between CE routers connected to the same PE router using BGP.

The configuration statements enable the VPN AB Router CE2 to communicate with the VPN A Router CE1 and the VPN B Router CE3, both directly connected to the Router PE1. VPN routes that originate from the remote PE routers (the PE2 Router, in this case) are placed in a global Layer 3 VPN routing table (bgp.l3vpn.inet.0) and routes with appropriate route targets are imported into the routing tables, as dictated by the VRF import policy configuration.

Figure 6: Configuration of Policy-Based Export for an Overlapping VPN



Configuring Router PE1

This section describes how to configure Router PE1 in the backbone entity for this overlapping VPN by means of policy-based export.

Configure the routing instances for VPN-A, VPN-AB, and VPN-B:

```
[edit]
routing-instances {
   VPN-A {
      instance-type vrf;
      interface fe-1/0/0.0;
      route-distinguisher 10.255.14.175:3;
      vrf-export A-out;
      vrf-import A-in;
      routing-options {
         auto-export;
         static {
            route 1.1.1.1/32 next-hop fe-1/0/0.0;
            route 1.1.1.2/32 next-hop fe-1/0/0.0;
```

```
        }
      }
    }
    VPN-AB {
      instance-type vrf;
      interface fe-1/1/0.0;
      route-distinguisher 10.255.14.175:9;
      vrf-export AB-out;
      vrf-import AB-in;
      routing-options {
        auto-export;
        static {
          route 1.1.3.1/32 next-hop fe-1/1/0.0;
          route 1.1.3.2/32 next-hop fe-1/1/0.0;
        }
      }
      VPN-B {
        instance-type vrf;
        interface fe-1/0/2.0;
        route-distinguisher 10.255.14.175:9;
        vrf-export B-out;
        vrf-import B-in;
        routing-options {
          auto-export;
          static {
            route 1.1.2.1/32 next-hop fe-1/0/2.0;
            route 1.1.2.2/32 next-hop fe-1/0/2.0;
          }
        }
      }
    }
  }
```

Configuring Router PE2

The configuration for Router PE2 is the same as that for Router PE1; however, the interface names might differ.

Example: Configuring Policy-Based Export for a Nonforwarding Instance

This example shows how to use the instance-import and instance-export statements to control route export between multiple instances. This is equivalent to using the vrf-import and vrf-export statements for VPNs, except these are with nonforwarding instances, not VRF instances.

There are two nonforwarding instances: data and voice. The following is the configuration for a PE router.

Configure the routing instances for data and voice:

```
[edit]
routing-instances {
  data {
    instance-type no-forwarding;
    interface t3-0/1/3.0;
    routing-options {
```

```
            instance-import data-import;
            auto-export;
            protocols {
               ospf {
                  export accept;
                  area 0.0.0.0 {
                     interface all;
                  }
               }
            }
         }
         voice {
            instance-type no-forwarding;
            interface t3-0/1/0.0;
            routing-options {
               instance-import voice-import;
               auto-export;
            }
            protocols {
               ospf {
                  export accept;
                  area 0.0.0.0 {
                     interface all;
                  }
               }
            }
         }
      }
   }
```

Configure a master policy:

```
[edit]
policy-options {
   policy-statement master-import {
      term a {
         from instance master;
         then {
            tag 11;
            accept;
         }
      }
      term b {
         from instance data;
         then {
            tag 10;
            accept;
         }
      }
   }
}
```

Configure policies for each instance:

```
[edit]
policy-options {
```

```
    policy-statement data-import {
        term a {
            from {
                instance master;
                tag 10;
                then accept;
            }
        }
        term b {
            then reject;
        }
    }
    policy-statement voice-import {
        term a {
            from {
                instance master;
                protocol ospf;
                tag 11;
            }
        }
        term b {
            then reject;
        }
    }
}
```

Configuring VRF Table Labels

You configure a separate label for each VRF to provide double lookup and egress filtering. To configure a label for a VRF, include the following statements:

```
[edit]
routing-instances {
    routing-instance-name {
        instance-type vrf;
        vrf-import [ policy-names ];
        vrf-export [ policy-names ];
        vrf-table-label;
    }
}
```

For more information about configuring VPNs, see the *JUNOS VPNs Configuration Guide*.

Configuring VRF Targets

Configuring a VPN routing and forwarding (VRF) target provides a configurable community within a VRF routing instance and allows a single policy for import and a single policy for export to replace the per-VRF policies for every community.

To configure a VRF target, include the vrf-target statement. Use the import and export options to specify the allowed communities to accept from neighbors and to send to neighbors:

```
vrf-target {
    community;
    export community-name;
    import community-name;
}
```

You can configure the statements at the following hierarchy levels:

- [edit routing-instances *routing-instance-name*]
- [edit logical-systems *logical-system-name* routing-instances *routing-instance-name*]

Within a hub-and-spoke configuration, you can configure a PE router not to advertise VPN routes from the primary (hub) instance. Instead, these routes are advertised from the secondary (downstream) instance. You can do this without configuring routing table groups, by using the no-vrf-advertise statement.

NOTE: This statement does not prevent the exportation of VPN routes to other VRF instances on the same router by configuring the [edit routing-options auto-export] statement.

To prevent advertising VPN routes from the primary instance, include the no-vrf-advertise statement:

```
no-vrf-advertise;
```

You can configure the statement at the following hierarchy levels:

- [edit routing-instances *routing-instance-name*]
- [edit logical-systems *logical-system-name* routing-instances *routing-instance-name*]

For more information about configuring VPNs, see the *JUNOS VPNs Configuration Guide*.

Configuring OSPF Domain IDs for VPNs

For most OSPF or OSPFv3 configurations involving Layer 3 VPNs, you do not need to configure an OSPF domain ID. However, for a Layer 3 VPN connecting multiple OSPF or OSPFv3 domains, configuring domain IDs can help you control LSA translation (for Type 3 and Type 5 LSAs) between the OSPF domains and back-door paths. The default domain ID is 0.0.0.0. Each VPN routing table in a PE router associated with an OSPF or OSPFv3 instance is configured with the same OSPF domain ID.

JUNOS Software is fully compliant with Internet draft draft-ietf-l3vpn-ospf-2547-04.txt, *OSPF as the Provider/Customer Edge Protocol for BGP/MPLS IP VPNs*.

For more detailed information about configuring VPNs, see the *JUNOS VPNs Configuration Guide*.

Without the domain IDs, there is no way to identify which domain the routes originated from after the OSPF or OSPFv3 routes are distributed into BGP routes and advertised across the BGP VPN backbone. Distinguishing which OSPF or OSPFv3 domain a route originated from allows classification of routes as Type 3 LSAs or Type 5 LSAs.

To configure a domain ID, perform the following tasks:

1. Specify a domain ID in the BGP extended community ID.
2. Set a route type.
3. Configure a VRF export policy to explicitly attach the outbound extended community ID to outbound routes.
4. Define a community with members that possess the community ID.

For more information about configuring export policies, see the *JUNOS Policy Framework Configuration Guide*.

This extended community ID can then be carried across the BGP VPN backbone. When the route is redistributed back as an OSPF or OSPFv3 route on the PE router and advertised to the CE near the destination, the domain ID identifies which domain the route originated from. The routing instance checks incoming routes for the domain ID. The route is then propagated as either a Type 3 LSA or Type 5 LSA.

When a PE router receives a route, it redistributes and advertises the route as either a Type 3 LSA or a Type 5 LSA, depending on the following:

- If the receiving PE router sees a Type 3 route with a matching domain ID, the route is redistributed and advertised as a Type 3 LSA.
- If the receiving PE router sees a Type 3 route without a domain ID (the extended attribute field of the route's BGP update does not include a domain ID), the route is redistributed and advertised as a Type 3 LSA.
- If the receiving PE router sees a Type 3 route with a non-matching domain ID, the route is redistributed and advertised as a Type 5 LSA.
- If the receiving PE router sees a Type 3 route with a domain ID, but the router does not have a domain ID configured, the route is redistributed and advertised as a Type 5 LSA.
- If the receiving PE router sees a Type 5 route, the route is redistributed and advertised as a Type 5 LSA, regardless of the domain ID.

On the local PE router, the prefix of the directly connected PE/CE interface is an active direct route. This route is also an OSPF or OSPFv3 route.

In the VRF export policy, the direct prefix is exported to advertise the route to the remote PE. This route is injected as an AS-External-LSA, much as when a direct route is exported into OSPF or OSPFv3.

Domain ID ensures that an originated summary LSA arrives at the remote PE as a summary LSA. Domain ID does not translate AS-external-LSAs into summary LSAs.

To configure an OSPF or OSPFv3 domain ID match condition for incoming Layer 3 VPN routes going into a routing instance, include the domain-id statement:

```
domain-id domain-id;
```

For *domain-id*, specify either an IP address or an IP address and a local identifier using the following format: *ip-address*:*local-identifier*. If you do not specify a local identifier with the IP address, the identifier is assumed to have a value of 0.

You can configure the statement at the following hierarchy levels:

- [edit routing-instances *routing-instance-name* protocols (ospf | ospf3)]
- [edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols (ospf | ospf3)]

If the router ID is not configured in the routing instance, the router ID is derived from an interface address belonging to the routing instance.

You can set a VPN tag for the OSPF or OSPFv3 external routes generated by the PE router. This prevents looping when a domain ID is used as an alternate route preference. By default, this tag is automatically calculated and needs no configuration. To configure the domain VPN tag for Type 5 LSAs, include the domain-vpn-tag *number* statement:

```
domain-vpn-tag number;
```

You can configure the statement at the following hierarchy levels:

- [edit routing-instances *routing-instance-name* protocols (ospf | ospf3)]
- [edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols (ospf | ospf3)]

The range is from 1 through 4,294,967,295. If you set VPN tags manually, you must set the same value for all PE routers in the VPN.

To set the route type, include the route-type-community statement:

```
route-type-community (iana | vendor);
```

You can include the statement at the following hierarchy levels:

- [edit routing-instances *routing-instance-name* protocols (ospf | ospf3)]
- [edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols (ospf | ospf3)]

The domain-id setting in the routing instance is for a match on inbound Layer 3 VPN routes. A VRF export policy must be explicitly set for the outbound extended community domain-id attribute. You must configure an export policy to attach the domain ID to outgoing routes. To configure an export policy to attach the domain ID and route distinguisher to the extended community ID on outbound routes, include the community statement:

```
policy-statement policy-name {
   term term-name {
      from protocol (ospf | ospf3);
      then {
         community add community-name;
         accept;
      }
   }
   term b {
      then reject;
   }
}
community community-name members [ target:target-id domain-id:domain-id];
```

You can include the statement at the following hierarchy levels:

- [edit policy-options policy-statement *policy-name* term *term-name* then]
- [edit logical-systems *logical-system-name* policy-options policy-statement *policy-name* term *term-name* then]

To define the members of a community, include the **community** statement:

```
community name {
   members [ community-ids ];
}
```

You can include the statement at the following hierarchy levels:

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

Examples: Configuring an OSPF Domain ID

Configure a domain ID as a match condition for inbound Layer 3 VPN routes. Then configure an export policy to tag the extended community ID and the route distinguisher onto outgoing routes:

```
[edit]
routing-instances {
   CE_A {
      instance-type vrf;
      interface ge-0/1/0.0;
      route-distinguisher 1:100;
      vrf-import vrf_import_routes;
      vrf-export vrf_export_routes;
      protocols {
         ospf {
            domain-id 1.1.1.1; # match for inbound routes
            route-type-community vendor;
            export vrf_import_routes;
            area 0.0.0.0 {
               interface ge-0/1/0.0;
            }
```

```
        }
      }
    }
  }
  policy-options {
    policy-statement vrf_export_routes {
      term a {
        from protocol ospf;
        then {
          community add export_target;
          accept;
        }
      }
      term b {
        then reject;
      }
    }
    community export_target members [ target:1:100 domain-id:1.1.1.1:0 ];
  }
```

Leak a noninstance route into the instance routing table:

```
[edit]
routing-options {
  interface-routes {
    rib-group inet inet_to_site_A;
  }
}
[edit]
rib-groups {
  inet_to_site_A {
    import-rib [ inet.0 site_A.inet.0 ];
  }
}
[edit]
protocols {
  ospf {
    rib-group inet_to_site_A;
  }
}
[edit]
policy-options {
  policy-statement announce_to_ce {
    term a {
      from {
        protocol direct;
        interface lo0.0;
      }
      then accept;
    }
  }
}
[edit]
routing-instances {
  site_A {
    protocols {
```

```
        ospf {
            export announce_to_ce;
        }
    }
}
}
```

Configuring Route Limits for Routing Tables

A route limit sets an upper limit for the number of paths and prefixes installed in routing tables. You can, for example, use a route limit to limit the number of routes received from the CE router in a VPN. A route limit applies only to dynamic routing protocols, not to static or interface routes.

To configure a route limit on route paths, include the maximum-paths statement:

maximum-paths *path-limit* <log-only | threshold *value* log-interval *seconds*>;

To configure a route limit on route prefixes, include the maximum-prefixes statement:

maximum-prefixes *prefix-limit* <log-only | threshold *value* log-interval *seconds*>;

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for these statements.

Specify the log-only option to generate warning messages only (an advisory limit). Specify the threshold option to generate warnings before the limit is reached. Specify the log-interval option to configure the minimum time interval between log messages.

There are two modes for route limits: advisory and mandatory. An advisory limit triggers warnings. A mandatory limit rejects additional routes after the limit is reached.

NOTE: Application of a route limit may result in unpredictable dynamic routing protocol behavior. For example, when the limit is reached and routes are rejected, BGP may not reinstall the rejected routes after the number of routes drops back below the limit. BGP sessions may need to be cleared.

For more information about configuring VPNs, see the *JUNOS VPNs Configuration Guide*.

Configuring Independent AS Domains

You can configure an independent autonomous system (AS) domain that is separate from the primary routing instance domain. An AS is a set of routers that are under a single technical administration and that generally use a single IGP and metrics to propagate routing information within the set of routers. An AS appears to other ASs to have a single, coherent interior routing plan and presents a consistent picture of what destinations are reachable through it.

Configuring an independent domain allows you to keep the AS paths of the independent domain from being shared with the AS path and AS path attributes of other domains, including the master routing instance domain.

If you are using BGP on the router, you must configure an AS number.

To configure an independent domain, include the independent-domain statement:

```
independent-domain;
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

There is a limit of 16 ASs for each domain.

Chapter 10

Summary of Routing Instances Configuration Statements

This chapter provides a reference for each of the routing instance configuration statements. The statements are organized alphabetically.

access-profile

Syntax access-profile *profile-name*;

Hierarchy Level [edit],
[edit routing-instances *routing-instances-name*],

Release Information Statement introduced in JUNOS Release 9.1.

Description Specify the access profile for use by the master routing instance.

Options *profile-name*—Name of the access profile.

Required Privilege Level access—To view this statement in the configuration.
access-control—To add this statement to the configuration.

description

Syntax description *text*;

Hierarchy Level [edit routing-instances *routing-instance-name*]

Release Information Statement introduced before JUNOS Release 7.4.

Description Provide a text description for the routing instance. If the text includes one or more spaces, enclose it in quotation marks (“ ”). Any descriptive text you include is displayed in the output of the show route instance detail command and has no effect on the operation of the routing instance.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Topics
- Complete Routing Instances Configuration Statements on page 229

forwarding-options

See *JUNOS Policy Framework Configuration Guide*

instance-type

Syntax instance-type (forwarding | l2vpn | layer2-control | no-forwarding | virtual-router | virtual-switch | vpls | vrf);

Hierarchy Level [edit logical-systems *logical-system-name* routing-instances *routing-instance-name*], [edit routing-instances *routing-instance-name*]

Release Information Statement introduced before JUNOS Release 7.4.
virtual-switch and layer2-control options introduced in JUNOS Release 8.4.

Description Define the type of routing instance.

Default no-forwarding

Options forwarding—Provide support for filter-based forwarding, where interfaces are not associated with instances. All interfaces belong to the default instance. Other instances are used for populating RPD learned routes. See "Configuring Filter-Based Forwarding" on page 258.

l2vpn—Provide support for Layer 2 VPNs. For more detailed information about configuring VPNs, see the *JUNOS VPNs Configuration Guide*.

layer2-control—(MX Series routers only) Provide support for RSTP or MSTP in customer edge interfaces of a VPLS routing instance. For more detailed information about configuring RSTP and MSTP, see the *JUNOS MX Series Ethernet Services Routers Layer 2 Configuration Guide*

no-forwarding—This is the default routing instance. Do not create a corresponding forwarding instance.

virtual-router—Similar to a VPN routing and forwarding instance type, but used for non-VPN-related applications. There are no VRF import, VRF export, VRF target, or route distinguisher requirements for this instance type.

virtual-switch—(MX Series routers only) Provide support for Layer 2 bridging. Use this routing instances type to isolate a LAN segment with its Spanning Tree Protocol (STP) instance and separates its VLAN identifier space. For more detailed information about configuring a virtual switch, see the *JUNOS MX Series Ethernet Services Routers Layer 2 Configuration Guide* and the *JUNOS MX Series Ethernet Services Routers Solutions Guide*.

vpls—Virtual private local-area network (LAN) service. Use this routing instance type for point-to-multipoint LAN implementations between a set of sites in a VPN. For more information about configuring VPLS, see the *JUNOS VPNs Configuration Guide*.

vrf—VPN routing and forwarding instance. Provides support for Layer 3 VPNs, where interface routes for each instance go into the corresponding forwarding table only. For more information about configuring VPNs, see the *JUNOS VPNs Configuration Guide*.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Specifying the Instance Type for Routing Instances on page 253
- *JUNOS VPNs Configuration Guide*

interface

Syntax interface *interface-name*;

Hierarchy Level [edit logical-systems *logical-system-name* routing-instances *routing-instance-name*], [edit routing-instances *routing-instance-name*]

Release Information Statement introduced before JUNOS Release 7.4.

Description Identify the logical, private interface between the provider edge (PE) router and the customer edge (CE) router on the PE side.

Options *interface-name*—Name of the interface.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring Routing Instances on page 251

no-vrf-advertise

Syntax no-vrf-advertise;

Hierarchy Level [edit logical-systems *logical-system-name* routing-instances *routing-instance-name*], [edit routing-instances *routing-instance-name*]

Release Information Statement introduced before JUNOS Release 7.4.

Description Prevent advertising VPN routes from a VRF instance to remote PEs.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring VRF Targets on page 265

protocols

Syntax

```
protocols {
    bgp {
        ... bgp-configuration ...
    }
    isis {
        ... isis-configuration ...
    }
    ldp {
        ... ldp-configuration ...
    }
    msdp {
        ... msdp-configuration ...
    }
    mstp {
        ... mstp-configuration ...
    }
    ospf {
        domain-id domain-id;
        domain-vpn-tag number;
        route-type-community (iana | vendor);
        ... ospf-configuration ...
    }
    ospf3 {
        domain-id domain-id;
        domain-vpn-tag number;
        route-type-community (iana | vendor);
        ... ospf3-configuration ...
    }
    pim {
        ... pim-configuration ...
    }
    rip {
        ... rip-configuration ...
    }
    ripng {
        ... ripng-configuration ...
    }
    rstp {
        rstp-configuration;
    }
    vstp {
        vstp configuration;
    }
}
```

Hierarchy Level [edit logical-systems *logical-system-name* routing-instances *routing-instance-name*], [edit routing-instances *routing-instance-name*]

Release Information Statement introduced before JUNOS Release 7.4.
Support for RIPng introduced in JUNOS Release 9.0.

Description Specify the protocol for a routing instance. You can configure multiple instances of the following supported protocols: BGP, IS-IS, LDP, MSDP, OSPF, OSPFv3, PIM, RIP, and RIPng.

Options bgp—Specify BGP as the protocol for a routing instance. For a description of the BGP configuration statements, see "BGP Configuration Guidelines" on page 721.

isis—Specify IS-IS as the protocol for a routing instance. For a description of the IS-IS configuration statements, see "IS-IS Configuration Guidelines" on page 321.

ldp—Specify LDP as the protocol for a routing instance. For more information about configuring LDP, see the *JUNOS MPLS Applications Configuration Guide*.

msdp—Specify the Multicast Source Discovery Protocol (MSDP) for a routing instance. For more information about configuring MSDP, see the *JUNOS Multicast Protocols Configuration Guide*.

mstp—Specify the Multiple Spanning Tree Protocol (MSTP) for a virtual switch routing instance. For more information about configuring MSTP, see the *JUNOS MX Series Ethernet Services Routers Layer 2 Configuration Guide*

ospf—Specify OSPF as the protocol for a routing instance. For a description of the OSPF configuration statements, see "OSPF Configuration Guidelines" on page 443.

ospf3—Specify OSPF version 3 (OSPFv3) as the protocol for a routing instance. For a description of the OSPFv3 configuration statements, see "OSPF Configuration Guidelines" on page 443.

NOTE: OSPFv3 supports the no-forwarding and vrf routing instance types only.

pim—Specify the Protocol Independent Multicast (PIM) protocol for a routing instance. For more information about configuring PIM, see the *JUNOS Multicast Protocols Configuration Guide*.

rip—Specify RIP as the protocol for a routing instance. For a description of the RIP configuration statements, see "RIP Configuration Guidelines" on page 583.

ripng—Specify RIP next generation (RIPng) as the protocol for a routing instance. For a description of the RIPng configuration statements, see "RIPng Configuration Guidelines" on page 633.

rstp—Specify the Rapid Spanning Tree Protocol (RSTP) for a virtual switch routing instance. For information about configuring RSTP, see the *JUNOS MX Series Ethernet Services Routers Layer 2 Configuration Guide*.

vstp—Specify the VLAN Spanning Tree Protocol (VSTP) for a virtual switch routing instance. for information about configuring VSTP, see the *JUNOS MX Series Ethernet Services Routers Layer 2 Configuration Guide*

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics

- Configuring Multiple Instances of BGP on page 240
- Configuring Multiple Instances of IS-IS on page 241
- Configuring Multiple Instances of LDP on page 245
- Configuring Multiple Instances of MSDP on page 246
- Configuring Multiple Instances of OSPF on page 247
- Configuring Multiple Instances of PIM on page 250
- Configuring Multiple Instances of RIP on page 251

route-distinguisher

Syntax route-distinguisher (*as-number*:*number* | *ip-address*:*number*);

Hierarchy Level [edit logical-systems *logical-system-name* routing-instances *routing-instance-name*], [edit routing-instances *routing-instance-name*]

Release Information Statement introduced before JUNOS Release 7.4.

Description Specify an identifier attached to a route, enabling you to distinguish to which VPN the route belongs. Each routing instance must have a unique route distinguisher associated with it. The route distinguisher is used to place bounds around a VPN so that the same IP address prefixes can be used in different VPNs without having them overlap. If the instance type is vrf, the route-distinguisher statement is required.

Options *as-number*:*number*—*as-number* is an assigned AS number and *number* is any 2-byte for 4-byte value. The AS number can be from 1 through 4,294,967,295. If the AS number is a 2-byte value, the administrative number is a 4-byte value. If the AS number is 4-byte value, the administrative number is a 2-byte value.

NOTE: In JUNOS Release 9.1 and later, the numeric range for AS numbers is extended to provide BGP support for 4-byte AS numbers, as defined in RFC 4893, *BGP Support for Four-octet AS Number Space*.

In JUNOS Release 9.2 and later, you can also configure a 4-byte AS number using the AS-dot notation format of two integer values joined by a period: < *16-bit high-order value in decimal* > . < *16-bit low-order value in decimal* > . For example, the 4-byte AS number of 65,546 in plain-number format is represented as 1.10 in the AS-dot notation format.

ip-address:*number*—*ip-address* is an IP address in your assigned prefix range (a 4-byte value) and *number* is any 2-byte value.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring Route Distinguishers for Routing Instances on page 257

routing-instances

Syntax routing-instances *routing-instance-name* { ... }

Hierarchy Level [edit],
[edit logical-systems *logical-system-name*]

Release Information Statement introduced before JUNOS Release 7.4.

Description Configure an additional routing entity for a router. You can create multiple instances of BGP, IS-IS, OSPF, OSPFv3, and RIP for a router. You can also create multiple routing instances for separating routing tables, routing policies, and interfaces for individual wholesale subscribers (retailers) in a Layer 3 wholesale network.

Default Routing instances are disabled for the router.

Options *routing-instance-name*—Name of the routing instance, a maximum of 128 characters. A routing instance name can contain letters, numbers, and hyphens.

NOTE: In JUNOS Release 9.6 and later, you can include a slash (/) in a routing-instance name only if a logical system is not configured. That is, you cannot include the slash character in a routing-instance name if a logical system other than the default is explicitly configured.

The remaining statements are explained separately.

NOTE: In JUNOS Release 9.0 and later, you cannot specify a routing-instance name of default or include special characters within the name of a routing instance.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics

- Routing Instances Configuration Guidelines on page 229
- Routing Instances Configuration Guidelines on page 229Example: Configuring E-LINE and E-LAN Services for a PBB Network on MX series Routers
- *JUNOS Policy Framework Configuration Guide*

routing-options

See routing-options

vrf-export

Syntax vrf-export [*policy-names*];

Hierarchy Level [edit logical-systems *logical-system-name* routing-instances *routing-instance-name*], [edit routing-instances *routing-instance-name*]

Release Information Statement introduced before JUNOS Release 7.4.

Description Define which routes are exported from a local instance table—*instance-name*.inet.0—to a remote PE router. Specify one or more policy names.

Default If the instance-type is vrf, vrf-export is a required statement. The default action is to reject.

Options *policy-names*—Specify one or more policy names.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring Secondary VRF Import and Export Policy on page 260

vrf-import

Syntax vrf-import [*policy-names*];

Hierarchy Level [edit logical-systems *logical-system-name* routing-instances *routing-instance-name*], [edit routing-instances *routing-instance-name*]

Release Information Statement introduced before JUNOS Release 7.4.

Description How routes are imported into the local PE router's VPN routing table—*instance-name*.inet.0—from the remote PE router.

Default If the instance-type is vrf, vrf-import is a required statement. The default action is to accept.

Options *policy-names*—Specify one or more policy names.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring Secondary VRF Import and Export Policy on page 260

vrf-table-label

Syntax vrf-table-label;

Hierarchy Level [edit logical-systems *logical-system-name* routing-instances *routing-instance-name*], [edit routing-instances *routing-instance-name*]

Release Information Statement introduced before JUNOS Release 7.4.

Description Enable mapping of the inner label of a packet to a specific VRF, thereby allowing the examination of the encapsulated IP header. All routes in the VRF configured with this option are advertised with the label allocated per VRF.

NOTE: This statement does not support IPv6 VPNs.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring VRF Table Labels on page 265

vrf-target

Syntax
```
vrf-target {
    community;
    import community;
    export community;
}
```

Hierarchy Level [edit logical-systems *logical-system-name* routing-instances *routing-instance-name*], [edit routing-instances *routing-instance-name*]

Release Information Statement introduced before JUNOS Release 7.4.

Description Configure a single policy for import and a single policy for export to replace the per-VRF policies for every community.

Options *community*—Community name.

import—Specifies the allowed communities to accept from neighbors.

export—Specifies the allowed communities to send to neighbors.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring VRF Targets on page 265

Part 4
Multitopology Routing

- Introduction to Multitopology Routing on page 287
- Multitopology Routing Configuration Guidelines on page 291
- Summary of Multitopology Routing Configuration Statements on page 303

Chapter 11

Introduction to Multitopology Routing

Multitopology Routing Overview

Multitopology Routing enables you to configure class-based forwarding for different types of traffic, such as voice, video, and data. Each type of traffic is defined by a topology that is used to create a new routing table for that topology. Multitopology Routing provides the ability to generate forwarding tables based on the resolved entries in the routing tables for the custom topologies you create. In this way, packets of different classes can be routed independently from one another.

This chapter discusses the following topics that provide background information about Multitopology Routing:

- Routing Table Naming Conventions for Multitopology Routing on page 287
- Routing Protocol Support for Multitopology Routing on page 288
- Filter-Based Forwarding Support on page 288

Routing Table Naming Conventions for Multitopology Routing

Each routing protocol creates a routing table based on the topology name, the instance name, and the purpose of the table. A routing table for each topology uses the following format:

logical-system-name/routing-instance-name:topology-name.protocol.identifier

The routing instance string is included only if the instance is not the master. The logical system string is included only if the logical system identifier has a value other than 0 (zero). Each routing table for a topology includes a colon (:) before the topology name that also separates the routing-instance name from the topology name. *protocol* is the protocol family, which can be inet or inet6. *identifier* is a positive integer that specifies the instance of the routing table. Table 6 on page 287 shows specific examples of routing tables for various topologies.

Table 6: Examples of Routing Tables for Custom Topologies

Name of Routing Table	Description
:voice.inet.0	Master instance, voice topology, unicast IPv4 routes
:voice.inet6.0	Master instance, voice topology, unicast IPv6 routes

Table 6: Examples of Routing Tables for Custom Topologies *(continued)*

Name of Routing Table	Description
:voice.inet.3	Master instance, voice topology, ingress label-switched paths (LSPs)
private_1/:voice.inet.0	Logical system private, voice topology, unicast IPv4 routes
customer-A:voice.inet.0	Virtual-router customer-A, voice topology, unicast IPV4 routes
customer-B:voice.inet.3	Virtual-router customer-B, voice topology, ingress LSPs
customer-A:voice.mpls.0	Virtual-router customer-A, voice topology, unicast carrier-of-carriers IPV4 routes

Routing Protocol Support for Multitopology Routing

To run Multitopology Routing, you must configure IP routing. Multitopology Routing supports OSPF version 2 (OSPFv2), static routes, and BGP. You must configure an interior gateway protocol (IGP), such as OSPFv2 or static routing. Configure BGP to add routes learned through BGP to the appropriate custom topologies.

OSPF in Multitopology Routing uses a single instance of OSPF to carry connectivity and IP reachability information for different topologies. That information is used to calculate shortest-path-first (SPF) trees and routing tables. OSPF in Multitopology Routing supports protocol extensions that include metrics that correspond to different topologies for link and prefix reachability information. The type-of-service (TOS) metric field is used to advertise the topology-specific metric for links and prefixes belonging to that topology. The TOS field is redefined as MT-ID in the payload of router, summary, and Type 5 and Type 7 autonomous-system-external link-state advertisements (LSAs).

BGP in Multitopology Routing provides the ability to resolve BGP routes against configured topologies. An inbound policy is used to select routes for inclusion in the appropriate routing tables for the topologies.

NOTE: Multitopology Routing is also supported on logical systems and the virtual router routing instance. No other routing instance type is supported on Multitopology Routing. For more information about configuring logical systems, see “Configuring Logical Systems” on page 135. For more information about configuring routing instances see, “Routing Instances Configuration Guidelines” on page 229. For more information about configure a virtual router instance, see the *JUNOS VPNs Configuration Guide*.

Filter-Based Forwarding Support

By default, the ingress interface forwards traffic to the default topology for each configured routing instance. Multitopology Routing supports filter-based forwarding, which enables you to match traffic on the ingress interface with a specific type of

forwarding class and then forward that traffic to the specified topology. You can further define how traffic is handled for each forwarding class by configuring additional firewall filters that match traffic for such values as the IP precedence field or the Differentiated Services code point (DSCP).

Multitopology Routing Standards

Multitopology Routing is defined in the following document:

- RFC 4915, *Multi-Topology (MT) Routing in OSPF*

Chapter 12

Multitopology Routing Configuration Guidelines

This chapter discusses the following tasks for configuring Multitopology Routing (MTR).

- Configuring Topologies on page 291
- Configuring Multitopology Routing in OSPF on page 292
- Configuring Multitopology Routing in Static Routes on page 298
- Configuring Multitopology Routing in BGP on page 299
- BGP Route Resolution in Multitopology Routing on page 299
- Configuring Filter-Based Forwarding for Multitopology Routing on page 300

Configuring Topologies

For Multitopology Routing to run on the router, you need to configure one or more topologies. For each topology, you specify a string value, such as voice, that defines the type of traffic, as well as an interface family, such as IPv4. In addition, a default topology is automatically created. You can also enable a topology for IPv4 multicast traffic. Each topology that you configure creates a new routing table and populates it with direct routes from the topology. For more information about the naming conventions for routing tables for topologies, see "Routing Table Naming Conventions for Multitopology Routing" on page 287. To configure a custom topology, include the following statements at the [edit routing options] hierarchy level:

```
[edit routing-options]
topologies {
    family (inet | inet6) {
        topology topology-name;
    }
}
```

Include the family inet statement to specify IPv4 traffic. Include the family inet6 statement to specify IPv6 traffic.

Include the topology *topology-name* statement to create a topology. For *topology-name*, specify a name for the topology in the form of a string. Typically, you would specify a name that describes the type of traffic, such as video. You can also specify ipv4-multicast to create a topology for IPv4 multicast traffic. A default topology is also automatically created.

Configuring Multitopology Routing in OSPF

Multitopology Routing OSPF (MT-OSPF) enables you to define multiple topologies and to configure topology-specific metrics for individual links as well as to exclude individual links from specific topologies. As a result, you can use a single instance of OSPF to carry connectivity and IP reachability information for different topologies. Information for different topologies is used to calculate independent shortest-path-first (SPF) trees and routing tables. For information about configuration tasks for MT-OSPF, see the following sections:

- Configuring Topologies and SPF Options for MT-OSPF on page 292
- Configuring a Prefix Export Limit for MT-OSPF on page 294
- Configuring a Topology to Appear Overloaded on page 294
- Configuring Interface Properties for MT-OSPF on page 294
- Disabling MT-OSPF on OSPF Interfaces on page 295
- Disabling MT-OSPF on Virtual Links on page 295
- Advertising MPLS Label-Switched Paths into MT-OSPF on page 296
- Configuring Other MT-OSPF Properties on page 297

Configuring Topologies and SPF Options for MT-OSPF

Include the following statements to enable topologies for OSPF and to configure topology identifiers. Any topologies you enable for OSPF must first be created under the [edit routing-options] hierarchy level. The routes for each topology are added to the routing table for the topology. For more information about the naming conventions for routing tables for topologies, see “Routing Table Naming Conventions for Multitopology Routing” on page 287.

The default topology is automatically created and has a topology identifier of 0 (zero), which cannot be modified. The routes that correspond to the default topology are added to the inet.0 routing table. You can, however, modify other parameters, such as shortest-path first (SPF) options. In addition, you can specify a topology for IPv4 multicast traffic. The topology for IPv4 multicast has a topology identifier of 1, which you cannot modify. The routes corresponding to this topology are added to the inet.2 routing table. You can also configure for each topology options for the SPF algorithm that override the default or globally configured SPF values. Include the following statements to configure a topology for OSPF and SPF options for the topology at the [edit protocols ospf] hierarchy level:

```
[edit protocols ospf]
topology (default | ipv4-multicast | name) {
   topology-id number;
   spf-options {
      delay milliseconds;
      holddown milliseconds;
      rapid-runs number;
   }
}
```

For *name*, include the name of a topology that you configured under the [edit routing-options] hierarchy level to create the topology.

Use ipv4-multicast for IPv4 multicast traffic. You must first enable this topology under the [edit routing-options] hierarchy level.

topology-id *number* is the topology identifier. The range for topology-id *number* is from 32 through 127 for any topology you create, except for the default and IPv4 multicast topologies. The identifier for those topologies is predefined and cannot be modified.

NOTE: Multitopology Routing is not currently supported for OSPF version 3 (OSPFv3).

You can configure SPF options for each topology. The values you configure for each of the following options override the default or globally configured values.

- The delay in the time between the detection of a topology change and when the SPF algorithm actually runs
- The maximum number of times that the SPF algorithm can run in succession before the hold-down timer begins
- The time to hold down, or wait, before running another SPF calculation after the SPF algorithm has run in succession the configured maximum number of times

To configure the SPF delay, include the delay statement when specifying the spf-options statement:

```
delay milliseconds;
```

By default, the SPF algorithm runs 200 milliseconds after the detection of a topology change. The range that you can configure is from 50 through 8000 milliseconds.

To configure the maximum number of times that the SPF algorithm can run in succession, include the rapid-runs statement when specifying the spf-options statement:

```
rapid-runs number;
```

The default number of SPF calculations that can occur in succession is 3. The range that you can configure is from 1 through 5. Each SPF algorithm is run after the configured SPF delay. When the maximum number of SPF calculations occurs, the hold-down timer begins. Any subsequent SPF calculation is not run until the hold-down timer expires.

To configure the SPF hold-down timer, include the holddown statement when specifying the spf-options statement:

```
holddown milliseconds;
```

The default is 5000 milliseconds, and the range that you can configure is from 2000 through 20,000 milliseconds. Use the hold-down timer to hold down, or wait, before running any subsequent SPF calculations after the SPF algorithm runs for the configured maximum number of times. If the network stabilizes during the hold-down

period and the SPF algorithm does not need to run again, the system reverts to the configured values for the delay and rapid-runs statements.

Configuring a Prefix Export Limit for MT-OSPF

By default, each topology uses the globally configured value to determine the maximum number of prefixes that can be exported into OSPF. You can override the globally configured value for any configured topology. Include the prefix-export-limit *number* statement at the [edit protocols ospf topology *name*] hierarchy level:

```
[edit protocols ospf]
topology (default | ipv4-multicast | name) {
   prefix-export-limit number;
}
```

The number that you can configure for each topology is from 0 through 4,294,967,295.

Configuring a Topology to Appear Overloaded

You can configure a specific topology so that it appears to be overloaded. You might do this when you want the topology to participate in OSPF routing but do not want it to be used for transit traffic.

To mark a topology as overloaded, include the overload statement:

```
[edit protocols ospf]
topology (default | ipv4-multicast | name) {
   overload;
}
```

Configuring Interface Properties for MT-OSPF

The default value of the topology metric is the same as the default metric value calculated by OSPF or the value configured for the OSPF metric. You can configure a topology-specific metric for an OSPF interface. To configure interfaces of MT-OSPF, Include the following statements at the [edit protocols ospf area *area-id*] hierarchy level:

```
interface interface-name {
   metric metric;
   topology (ipv4-multicast | name);
      metric metric;
   }
}
```

All OSPF interfaces have a cost, which is a routing metric that is used in the link-state calculation. Routes with lower total path metrics are preferred over those with higher path metrics. The default value for the OSPF metric for an interface is 1. You can modify the default value for an OSPF interface and configure a topology-specific metric for that interface. The topology-specific metric applies to routes advertised from the interface that belong only to that topology. The range that you can configure is from 1 through 65,535.

You can also configure any interface that belongs to one or more topologies to advertise the direct interface addresses without actually running OSPF on that interface. By default, OSPF must be configured on an interface for direct interface addresses to be advertised as interior routes. Include the passive statement at the [edit protocols ospf area *area-id* interface *interface-name*] hierarchy level:

```
[edit protocols ospf]
area area-id {
   interface interface-name {
      passive;
      topology name;
   }
}
```

NOTE: If you configure an interface with the passive statement, it applies to all the topologies to which the interface belongs. You cannot configure an interface as passive for only one specific topology and have it remain active for any other topologies to which it belongs.

Disabling MT-OSPF on OSPF Interfaces

By default, all topologies configured for OSPF are enabled on all OSPF interfaces. You can disable one or more configured topologies on an OSPF interface. To disable a configured topology on an OSPF interface, include the disable statement at the [edit protocols ospf area *area-id* interface *interface-name* topology *name*] hierarchy level:

```
[edit protocols ospf]
area area-id {
   interface interface-name {
      topology (ipv4-multicast | name) {
         disable;
      }
   }
}
```

You cannot disable an interface in the default topology and have it remain active in any other configured topologies.

NOTE: If you disable OSPF on an interface by including the disable statement at the [edit protocols ospf area *area-id* interface *interface-name*] hierarchy level, the interface is disabled for all topologies, including the default topology.

Disabling MT-OSPF on Virtual Links

By default, control packets sent to the remote end of a virtual link must be forwarded using the default topology. In addition, the transit area path consists only of links that are in the default topology. You can disable a virtual link for a configured topology, but not for a default topology. Include the disable statement at the [edit

protocols ospf area *area-id* virtual-link neighbor-id *router-id* transit-area *area-id* topology *name*] hierarchy level:

```
[edit protocols ospf]
area area-id {
   virtual-link neighbor-id router-id transit-area area-id {
      topology (ipv4-multicast | name) {
         disable;
      }
   }
}
```

NOTE: If you disable the virtual link by including the disable statement at the [edit protocols ospf area *area-id* virtual-link neighbor-id *router-id* transit-area *area-id*] hierarchy level, you disable the virtual link for all topologies, including the default topology. You cannot disable the virtual link only in the default topology.

Advertising MPLS Label-Switched Paths into MT-OSPF

You can advertise label-switched paths (LSPs) into OSPFv2 as point-to-point links so that all participating routers can take the LSP into account when performing SPF calculations. By default, all topologies configured for OSPF are enabled on all MPLS LSPs advertised into OSPF. You can override this behavior by disabling one or more configured topologies on an MPLS LSP.

The LSP advertisement contains a local address (the from address of the LSP), a remote address (the to address of the LSP), and a metric with the following precedence:

1. Use the LSP metric defined under OSPFv2.
2. Use the LSP metric configured under MPLS.
3. If you do not configure any of the above, use the default OSPFv2 metric of 1.

In addition, the default value of the topology-specific metric is the same as the default metric calculated by OSPF or configured for the MPLS LSPs. You can also override this value by configuring a specific metric for the topology. For more information about configuring a topology-specific metric, see "Configuring Interface Properties for MT-OSPF" on page 294.

To disable a topology on LSPs and configure a label-switched path metric for OSPFv2, include the following statements at the [edit protocols ospf] hierarchy level:

```
[edit protocols ospf]
area area-id {
   label-switched-path name;
      metric metric;
      topology (ipv4-multicast | name) {
         disable;
      }
   }
}
```

NOTE: You cannot disable an MPLS LSP only on the default topology and have it remain enabled on other topologies.

For more information about advertising LSPs, see the *JUNOS MPLS Applications Configuration Guide*.

Configuring Other MT-OSPF Properties

You can also configure the following properties for all topologies in an instance. You cannot configure the following properties for an individual topology:

- Disable not-so-stubby-area (NSSA) support on an autonomous-system border router (ASBR)
- Modify the preference value for OSPF internal routes
- Modify the default preference value for OSPF external routes
- Modify the reference-bandwidth value
- Enable graceful restart

To disable exporting Type 7 LSAs into LSAs, include the no-nssa-abr statement.

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

```
[edit protocols ospf]
no-nssa-abr;
```

By default, internal OSPF routes have a preference value of 10, and external OSPF routes have a preference value of 150. To modify the preference values for all topologies, include the preference statement (for internal routes) or the external-preference statement (for external routes):

```
[edit protocols ospf]
external-preference preference;
preference preference;
```

For a complete list of hierarchy levels at which you can configure these statements, see the statement summary sections for these statements.

You can configure a preference value of from 0 through 255 for each statement.

The reference bandwidth is used to calculate the default cost of a route using the following formula:

```
cost = reference-bandwidth / bandwidth
```

The default value for the reference bandwidth is 100 Mbps (which you specify as 100,000,000), which gives a metric of 1 for any bandwidth that is 100 Mbps or greater. To modify the default value, Include the reference-bandwidth statement:

```
[edit protocols ospf]
```

reference-bandwidth;

The range that you can specify is from 9,600 through 1,000,000,000,000

For a complete list of hierarchy levels at which you can configure this statement, see the statement summary section for this statement.

NOTE: You can specify topology-specific metrics for routes advertised from an interface. For more information, see "Configuring Interface Properties for MT-OSPF" on page 294.

Graceful restart enables a restarting router and its neighbors to continue forwarding packets without disrupting network performance. Because neighboring routers assist in the restart (these neighbors are called *helper routers*), the restarting router can quickly resume full operation without recalculating algorithms.

Graceful restart is disabled by default. You can globally configure graceful restart for all routing protocols at the [edit routing-options] hierarchy level. To configure graceful restart parameters specifically for OSPF, include the graceful-restart statement at the [edit protocols ospf] hierarchy level. For more information about how to configure graceful restart, see the *JUNOS High Availability Configuration Guide*.

Configuring Multitopology Routing in Static Routes

You can configure static routes to become installed in the routing table for any configured topology. Include the rib *routing-table-name* statement at the [edit routing-options] hierarchy level:

```
[edit routing-options]
   rib routing-table-name {
      static {
         route destination-prefix {
            next-hop;
         }
         static-options;
      }
   }
}
```

For *routing-table-name*, use the following format: *logical-system-name/routing-instance-name*:*topology-name*.*protocol*.*identifier*. The routing instance string is included only if the instance is not the master. The logical system string is included only if the logical system identifier has a value other than 0 (zero). Each routing table for a topology includes a colon (:) before the topology name that also separates the routing instance name from the topology name. *protocol* is the protocol family, which can be inet or inet6. *identifier* is a positive integer that specifies the instance of the routing table. When you create a topology for an instance (master or virtual-router), a new routing table is created within the instance for that topology. For more detailed information about routing table naming conventions for Multitopology Routing, see "Routing Table Naming Conventions for Multitopology Routing" on page 287.

For route *destination-prefix*, specify the destination of the route in the following way: *network/mask-length*, where *network* is the network portion of the IP address and *mask-length* is the destination prefix length. You can specify an IPv4 or IPv6 address.

You can optionally specify how to reach the destination by including the *next-hop* statement.

In addition, you can specify *static-options*, which defines additional information about static routes that is included with the route when it is installed in the routing table. For more information about specific static options you can optionally configure, see "Configuring Static Route Options" on page 65.

Configuring Multitopology Routing in BGP

Multitopology Routing in BGP enables you to configure a community target identifier for each type of traffic, or topology. The target community identifies the destination to which the route is going. BGP uses these community target identifiers to have routes imported into the routing tables for the specific topologies. The forwarding class then determines which table to use to forward traffic.

To configure Multitopology Routing in BGP, include the community target *identifier* statement at the [edit protocols family (inet | inet6) unicast topology *name*] hierarchy level:

```
[edit protocols bgp]
family (inet | inet6) {
   unicast {
      topology name {
         community target identifier;
      }
   }
}
```

Multitopology Routing in BGP is also supported for BGP groups and BGP peers. To configure for a BGP group, include the family (inet | inet6) unicast topology *name* community target *identifier* statement at the [edit protocols bgp group *group-name*] hierarchy level. To configure for a BGP peer, include the family (inet | inet6) unicast topology *name* community target *identifier* statement at the [edit protocols bgp group *group-name* neighbor *address*] hierarchy level.

BGP Route Resolution in Multitopology Routing

The default behavior is for the JUNOS Software to resolve BGP routes against the inet.0 and inet.3 routing tables. By default, the secondary route points to the next hop of the primary BGP route. This means that under the default behavior, BGP cannot perform secondary route resolution. Multitopology Routing in BGP provides support for secondary routes to resolve to an independent set of next hops.

When Multitopology Routing in BGP resolves a route against the inet.0 routing table, a forwarding state is generated to match the topologies for which you configured a BGP import policy.

Configuring Filter-Based Forwarding for Multitopology Routing

Each routing instance (master or virtual-router) supports one default topology to which all forwarding classes are forwarded. For Multitopology Routing, you can configure a firewall filter on the ingress interface to match a specific forwarding class, such as expedited forwarding, with a specific topology. The traffic that matches the specified forwarding class is then added to the routing table for that topology.

To configure filter-based forwarding for Multitopology Routing, include the following statements at the [edit firewall] hierarchy level:

```
[edit firewall]
family (inet | inet6) {
    filter filter-name {
        term term-name {
            from {
                forwarding-class (assured-forwarding | best-effort | expedited-forwarding |
                    network-control)
            }
            then {
                (topology topology-name | routing-instance routing-instance-name topology
                    topology-name | logical-system logical-system-name topology topology-name
                    | logical-system logical-system-name routing-instance routing-instance-name
                    topology topology-name);
            }
        }
    }
}
```

To configure the family address type, specify family inet to filter IPv4 packets or family inet6 to filter IPv6 packets.

To configure the filter name, include the filter *filter-name* statement. The filter name can contain letters, numbers, and hyphens (-) and can be up to 64 characters long. To include spaces in the name, enclose the entire name in quotation marks (“ ”).

Each filter consists of one or more terms. To configure a term, include the term *term-name* statement. The term name can contain letters, numbers, and hyphens (-) and can be up to 255 characters long. To include spaces in the name, enclose the entire name in quotation marks (“ ”). Each term name must be unique within a filter.

Include the forwarding-class *class* statement to define the forwarding class against which to match the incoming packets. You can configure the following types of forwarding classes: assured-forwarding, expedited-forwarding, best-effort, and network-control.

You can specify multiple terms in a filter, effectively chaining together a series of match-action operations to apply to the packets on an interface. Firewall filter terms are evaluated in the order in which you specify them in the configuration. To reorder terms, use the configuration mode insert command. For example, the command insert term up before term start places the term up before the term start.

Use the topology statement to specify that packets that match the specified forwarding class be directed to the specified topology.

For a topology in the master instance, include the topology *name* statement, where *name* is the name of the topology.

For a topology in a nonmaster instance, include the routing-instance *routing-instance-name* topology *topology-name* statement, where *routing-instance-name* is the name of the routing instance and *topology-name* is the name of the topology.

For a topology in a nonmaster logical system, include the logical-system *logical-system-name* topology *topology-name* statement, where *logical-system-name* is the name of the logical system and *topology-name* is the name of the topology.

For a topology in a nonmaster instance within a nonmaster logical system, include the logical-system *logical-system-name* routing-instance *routing-instance-name* topology *topology-name* statement, where *logical-system-name* is the name of the logical system, *routing-instance-name* is the name of the routing instance configured within the logical system, and *topology-name* is the name of the topology.

You must apply the filter to an ingress interface. Include the following statements to apply the filter to an interface:

```
[edit interfaces interface-name]
unit number {
    family (inet | inet6) {
        filter {
            input filter-name {
        }
    }
}
```

For more detailed information about how to configure firewall filters, see the *JUNOS Policy Framework Configuration Guide*.

Chapter 13

Summary of Multitopology Routing Configuration Statements

The following sections explain each of the Multitopology Routing configuration statements. They are organized alphabetically.

community

Syntax

```
community {
    target identifier;
}
```

Hierarchy Level

[edit logical-systems *logical-system-name* protocols bgp family (inet | inet6) unicast topology *name*],
[edit logical-systems *logical-system-name* protocols bgp group *group-name* family (inet | inet6) unicast topology *name*],
[edit logical-systems *logical-system-name* protocols bgp group *group-name* neighbor *address* family (inet | inet6) unicast topology *name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp family (inet | inet6) unicast topology *name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name* family (inet | inet6) unicast topology *name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address* family (inet | inet6) unicast topology *name*],
[edit protocols bgp family (inet | inet6) unicast topology *name*],
[edit protocols bgp group *group-name* family (inet | inet6) unicast topology *name*],
[edit protocols bgp group *group-name* neighbor *address* family (inet | inet6) unicast topology *name*],
[edit routing-instances *routing-instance-name* protocols bgp family (inet | inet6) unicast topology *name*],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name* family (inet | inet6) unicast topology *name*],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address* family (inet | inet6) topology *name*]

Release Information

Statement introduced in JUNOS Release 9.0.

Description

Configure the community to identify the multitopology routes. BGP uses the target community identifier to install the routes it learns in the appropriate Multitopology Routing tables.

Options

target *identifier*—Configure the destination to which the route is going.

Required Privilege Level

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

Related Topics

- Configuring Multitopology Routing in BGP on page 299

rib

Syntax

```
rib routing-table-name {
    static {
        route destination-prefix {
            next-hop;
        }
        static-options;
    }
}
```

Hierarchy Level

[edit logical-systems *logical-system-name* routing-options],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options],
[edit routing-instances *routing-instance-name* routing-options],
[edit routing-options]

Release Information

Statement support for Multitopology Routing introduced in JUNOS Release 9.0.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description

Configure a static route to install routes in the routing table for a specific topology.

Options

routing-table-name—Name of the routing table for a topology. Use the following format: *logical-system-name/routing-instance-name*:*topology-name*.*protocol*.*identifier*. Include the routing instance string only if the instance is not the master. The logical system string is included only if the logical system identifier has a value other than 0 (zero). Each routing table for a topology includes a colon (:) before the topology name. *protocol* is the protocol family, which can be inet or inet6. *identifier* is the positive integer that specifies the instance of the routing table. For example, to install IPv6 routes to the routing table for a topology named voice in the master instance, include :voice.inet6.0.

The remaining statements are explained separately.

Required Privilege Level

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

Related Topics

- static
- Configuring Multitopology Routing in Static Routes on page 298

topologies

Syntax

```
topologies {
    family (inet | inet6) {
        topology topology-name;
    }
}
```

Hierarchy Level

[edit logical-systems *logical-system-name* routing-options],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options],
[edit routing-instances *routing-instance-name* routing-options],
[edit routing-options]

Release Information

Statement introduced in JUNOS Release 9.0.

Description

Configure a topology for Multitopology Routing. Each topology creates a new routing table that is populated with direct routes from the topology.

Options

family—Configure the type of family address type.

inet—IPv4

inet6—IPv6

The remaining statement is explained separately.

Required Privilege Level

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

Related Topics

- topology (Multitopology Routing)
- Configuring Topologies on page 291

topology

- topology (Filter-Based Forwarding) on page 308
- topology (Multitopology Routing) on page 309
- topology (OSPF) on page 310
- topology (OSPF Interface) on page 311

topology (Filter-Based Forwarding)

Syntax topology *topology-name*;

Hierarchy Level [edit firewall family (inet | inet6) filter *filter-name* term *term-name* then],
[edit firewall family (inet | inet6) filter *filter-name* term *term-name* then logical-system *logical-system-name*],
[edit firewall family (inet | inet6) filter *filter-name* term *term-name* then logical-system *logical-system-name* routing-instance *routing-instance-name*],
[edit firewall family (inet | inet6) filter *filter-name* term *term-name* then routing-instance *routing-instance-name*]

Release Information Statement introduced in JUNOS Release 9.0.

Description Configure a topology for filter-based forwarding for Multitopology Routing. The firewall filter you apply to the ingress interface is used to look up traffic against the configured topology, and, if a route matches the conditions you configure for the term, the route is accepted and added to the to the routing table for the specific topology.

There are multiple ways to configure a topology for filter-based forwarding, depending on the type of instance or logical system you want to specify for the forwarding class. See Options for more information.

NOTE: The options for logical system and routing instance precede the topology statement with the then statement.

Options *topology-name*—Name of a topology against which you want to match traffic.

logical-system *logical-system-name* topology *topology-name*—For a nonmaster logical system, specify the name of the logical system and a topology name configured for a nonmaster logical system.

routing-instance *routing-instance-name* topology *topology-name*—For a nonmaster routing instance, specify the name of the routing instance and a topology name configured for a nonmaster routing instance.

logical-system *logical-system-name* routing-instance *routing-instance-name* topology *topology-name*—For a nonmaster routing instance configured within a nonmaster logical system, specify the name of the logical system, the name of the routing instance, and a topology name configured for a nonmaster routing instance within a nonmaster logical system.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring Filter-Based Forwarding for Multitopology Routing on page 300
- *JUNOS Policy Framework Configuration Guide*

topology (Multitopology Routing)

Syntax topology *topology-name*;

Hierarchy Level [edit logical-systems *logical-system-name* routing-options topologies family (inet | inet6)],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options topologies family (inet | inet6)],
[edit routing-instances *routing-instance-name* routing-options topologies family (inet | inet6)],
[edit routing-options topologies family (inet | inet6)]

Release Information Statement introduced in JUNOS Release 9.0.

Description Configure the name of a topology configured to run Multitopology Routing.

Options *topology-name*—Name of the topology. Include a string value that describes the type of traffic, such as voice or video. For IPv4 multicast traffic, include ipv4-multicast as the name.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- topologies
- Configuring Topologies on page 291

topology (OSPF)

Syntax

```
topology (default | ipv4-multicast | name) {
    topology-id number;
    spf-options {
        delay milliseconds;
        holddown milliseconds;
        rapid-runs number;
    }
}
```

Hierarchy Level

[edit logical-systems *logical-system-name* protocols ospf],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ospf],
[edit protocols ospf],
[edit routing-instances *routing-instance-name* protocols ospf]

Release Information

Statement introduced in JUNOS Release 9.0.

Description

Enable a topology for OSPF Multitopology Routing. You must first configure one or more topologies under the [edit routing-options] hierarchy level.

Options

default—Name of the default topology. This topology is automatically created and all routes that correspond to it are automatically added to the inet.0 routing table. You can modify certain default parameters, such as for the shortest-path-first (SPF) algorithm.

ipv4-multicast—Name of the topology for IPv4 multicast traffic.

name—Name of a topology you configured under the [edit routing-options] hierarchy level to create a topology for a specific type of traffic, such as voice or video.

Required Privilege Level

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

Related Topics

- Configuring Topologies and SPF Options for MT-OSPF on page 292

topology (OSPF Interface)

Syntax
```
topology (ipv4-multicast | name) {
    metric metric;
}
```

Hierarchy Level
```
[edit logical-systems logical-system-name protocols ospf area area-id interface
    interface-name],
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols
    ospf area area-id interface interface-name],
[edit protocols ospf area area-id interface interface-name],
[edit routing-instances routing-instance-name protocols ospf area area-id interface
    interface-name]
```

Release Information Statement introduced in JUNOS Release 9.0.

Description Configure interface-specific properties for MT-OSPF, including topology-specific metric values for an interface.

Default The default value of the topology metric is the same as the default metric value calculated by OSPF or the value configured for the OSPF metric.

Options ipv4-multicast—Name of the topology for IPv4 multicast traffic.

name—Name of a topology created under the [edit routing-options] hierarchy level.

metric *metric*—Cost of a route from an OSPF interface. You can specify a metric value for a topology that is different from the value specified for the interface.
Range: 1 through 65,535
Default: 1

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring Interface Properties for MT-OSPF on page 294

topology-id

Syntax topology-id *number*;

Hierarchy Level [edit logical-systems *logical-system-name* protocols ospf topology *name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ospf topology *name*],
[edit protocols ospf topology *name*],
[edit routing-instances *routing-instance-name* protocols ospf topology *name*]

Release Information Statement introduced in JUNOS Release 9.0.

Description Configure a topology identifier for a topology enabled for OSPF.

Default The default identifier for the default topology is 0, and the default identifier for the topology for IPv4 multicast traffic is 1. These identifiers are predefined and cannot be modified.

Options *number*—the integer value used to identify the topology.
Range: 32 through 127

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- topology (OSPF)
- Configuring Topologies and SPF Options for MT-OSPF on page 292

Part 5
Interior Gateway Protocols

- Introduction to IS-IS on page 315
- IS-IS Configuration Guidelines on page 321
- Summary of IS-IS Configuration Statements on page 373
- Introduction to OSPF on page 431
- OSPF Configuration Guidelines on page 443
- Summary of OSPF Configuration Statements on page 499
- Introduction to RIP on page 581
- RIP Configuration Guidelines on page 583
- Summary of RIP Configuration Statements on page 605
- Introduction to RIPng on page 631
- RIPng Configuration Guidelines on page 633
- Summary of RIPng Configuration Statements on page 643
- Introduction to ICMP Router Discovery on page 659
- ICMP Router Discovery Configuration Guidelines on page 661
- Summary of ICMP Router Discovery Configuration Statements on page 665
- Introduction to Neighbor Discovery on page 675
- Neighbor Discovery Configuration Guidelines on page 677
- Summary of Neighbor Discovery Router Advertisement Configuration Statements on page 685
- Secure Neighbor Discovery Configuration Guidelines on page 699
- Summary of Secure Neighbor Discovery Configuration Statements on page 703

Chapter 14

Introduction to IS-IS

This chapter discusses the following topics that provide background information about IS-IS:

- IS-IS Overview on page 315
- IS-IS Extensions to Support Traffic Engineering on page 317
- IS-IS Extensions to Support Route Tagging on page 318
- IS-IS Standards on page 318

IS-IS Overview

IS-IS protocol is an interior gateway protocol (IGP) that uses link-state information to make routing decisions.

IS-IS is a link-state IGP that uses the shortest path first (SPF) algorithm to determine routes. IS-IS evaluates the topology changes and determines whether to perform a full SPF recalculation or a partial route calculation (PRC). This protocol originally was developed for routing International Organization for Standardization (ISO) Connectionless Network Protocol (CLNP) packets.

NOTE: Because IS-IS uses ISO addresses, the configuration of IP version 6 (IPv6) and IP version 4 (IPv4) implementations of IS-IS is identical.

This section discusses the following topics:

- IS-IS Terminology on page 315
- ISO Network Addresses on page 316
- IS-IS Packets on page 317
- Persistent Route Reachability on page 317

IS-IS Terminology

An IS-IS network is a single autonomous system (AS), also called a *routing domain*, that consists of *end systems* and *intermediate systems*. End systems are network entities that send and receive packets. Intermediate systems send and receive packets and relay (forward) packets. (Intermediate system is the Open System Interconnection [OSI] term for a router.) ISO packets are called network *protocol data units* (*PDUs*).

In IS-IS, a single AS can be divided into smaller groups called *areas*. Routing between areas is organized hierarchically, allowing a domain to be administratively divided into smaller areas. This organization is accomplished by configuring *Level 1* and *Level 2* intermediate systems. Level 1 systems route within an area; when the destination is outside an area, they route toward a Level 2 system. Level 2 intermediate systems route between areas and toward other ASs.

ISO Network Addresses

IS-IS uses ISO network addresses. Each address identifies a point of connection to the network, such as a router interface, and is called a *network service access point* (*NSAP*).

IS-IS supports multiple NSAP addresses on the loopback (lo0) interface.

An end system can have multiple NSAP addresses, in which case the addresses differ only by the last byte (called the *n-selector*). Each NSAP represents a service that is available at that node. In addition to having multiple services, a single node can belong to multiple areas.

Each network entity also has a special network address called a *network entity title* (*NET*). Structurally, an NET is identical to an NSAP address but has an n-selector of 00. Most end systems and intermediate systems have one NET. Intermediate systems that participate in multiple areas can have multiple NETs.

The following ISO addresses illustrate the IS-IS address format:

```
49.0001.00a0.c96b.c490.00
49.0001.2081.9716.9018.00
```

The first portion of the address is the area number, which is a variable number from 1 through 13 bytes. The first byte of the area number (49) is the authority and format indicator (AFI). The next bytes are the assigned domain (area) identifier, which can be from 0 through 12 bytes. In the examples above, the area identifier is 0001.

The next six bytes form the system identifier. The system identifier can be any six bytes that are unique throughout the entire domain. The system identifier commonly is the media access control (MAC) address (as in the first example, 00a0.c96b.c490) or the IP address expressed in binary-coded decimal (BCD) (as in the second example, 2081.9716.9018, which corresponds to IP address 208.197.169.18). The last byte (00) is the n-selector.

NOTE: The system identifier cannot be 0000.0000.0000. All 0s is an illegal setting and the adjacency is not formed with this setting.

To provide help with IS-IS debugging, the JUNOS Software supports dynamic mapping of ISO system identifiers to the hostname. Each system can be configured with a hostname, which allows the system identifier-to-hostname mapping to be carried in a dynamic hostname type length value (TLV) in IS-IS link-state protocol data units (LSPs). This permits ISs in the routing domain to learn about the ISO system identifier of a particular IS.

IS-IS Packets

IS-IS uses the following protocol data units (PDUs) to exchange protocol information:

- IS-IS hello (IIH) PDUs—Broadcast to discover the identity of neighboring IS-IS systems and to determine whether the neighbors are Level 1 or Level 2 intermediate systems.
- Link-state PDUs —Contain information about the state of adjacencies to neighboring IS-IS systems. Link-state PDUs are flooded periodically throughout an area.
- Complete sequence number PDUs (CSNPs)—Contain a complete list of all link-state PDUs in the IS-IS database. CSNPs are sent periodically on all links, and the receiving systems use the information in the CSNP to update and synchronize their link-state PDU databases. The designated router multicasts CSNPs on broadcast links in place of sending explicit acknowledgments for each link-state PDU .
- Partial sequence number PDUs (PSNPs)—Multicast by a receiver when it detects that it is missing an link-state PDU; that is, when its link-state PUD database is out of date. The receiver sends a PSNP to the system that transmitted the CSNP, effectively requesting that the missing link-state PDU be transmitted. That router, in turn, forwards the missing link-state PDU to the requesting router.

Persistent Route Reachability

IPv4 and IPv6 route reachability information in IS-IS link-state PDUs is preserved when you commit a configuration. IP prefixes are preserved to their original packet fragment upon LSP regeneration.

IS-IS Extensions to Support Traffic Engineering

To help provide traffic engineering and MPLS with information about network topology and loading, extensions have been added to the JUNOS implementation of IS-IS. Specifically, IS-IS supports new TLVs that specify link attributes. These TLVs are included in the IS-IS link-state PDUs. The link-attribute information is used to populate the traffic engineering database, which is used by the Constrained Shortest Path First (CSPF) algorithm to compute the paths that MPLS LSPs take. This path information is used by RSVP to set up LSPs and reserve bandwidth for them.

NOTE: Whenever possible, use IS-IS IGP shortcuts instead of traffic engineering shortcuts.

The traffic engineering extensions are defined in Internet draft draft-isis-traffic-traffic-02, *IS-IS Extensions for Traffic Engineering*.

IS-IS IGP Shortcuts

In IS-IS, you can configure shortcuts, which allow IS-IS to use an LSP as the next hop as if it were a subinterface from the ingress routing device to the egress routing device. The address specified on the **to** statement at the [**edit protocols mpls label-switched-path** *lsp-path-name*] hierarchy level must match the router ID of the egress routing device for the LSP to function as a direct link to the egress routing device and to be used as input to IS-IS SPF calculations. When used in this way, LSPs are no different than Asynchronous Transfer Mode (ATM) and Frame Relay virtual circuits (VCs), except that LSPs carry only IPv4 traffic.

IS-IS Extensions to Support Route Tagging

To control the transmission of routes into IS-IS, or to control transmission of IS-IS routes between different IS-IS levels, you can tag routes with certain attributes. IS-IS routes can carry these attributes, which the routing policies can use to export and import routes between different IS-IS levels. A sub-TLV to the IP prefix TLV is used to carry the tag or attribute on the routes.

NOTE: Route tagging does not work when IS-IS traffic engineering is disabled.

IS-IS Standards

IS-IS is defined in the following documents:

- ISO 8473, *Protocol for providing the connectionless-mode network services*
- ISO 9542, *End System to Intermediate System Routing Exchange Protocol for Use in Conjunction with the Protocol for the Provision of the Connectionless-mode Network Service*
- ISO 10589, *Intermediate System to Intermediate System Routing Exchange Protocol for Use in Conjunction with the Protocol for the Provision of the Connectionless-mode Network Service*
- RFC 1195, *Use of OSI IS-IS for Routing in TCP/IP and Dual Environments*
- RFC 2973, *IS-IS Mesh Groups*
- RFC 3787, *Recommendations for Interoperable IP Networks Using Intermediate System to Intermediate System (IS-IS)*
- RFC 5120, *M-ISIS: Multi Topology (MT) Routing in Intermediate System to Intermediate Systems (IS-ISs)*
- RFC 5130, *A Policy Control Mechanism in IS-IS Using Administrative Tags*
- RFC 5301, *Dynamic Hostname Exchange Mechanism for IS-IS*
- RFC 5302, *Domain-wide Prefix Distribution with Two-Level IS-IS*

- RFC 5303, *Three-Way Handshake for IS-IS Point-to-Point Adjacencies*
- RFC 5304, *IS-IS Cryptographic Authentication*
- RFC 5305, *IS-IS Extensions for Traffic Engineering*
- RFC 5306, *Restart Signaling for IS-IS*
- RFC 5307, *IS-IS Extensions in Support of Generalized Multi-Protocol Label Switching (GMPLS)*
- RFC 5308, *Routing IPv6 with IS-IS*
- RFC 5309, *Point-to-Point Operation over LAN in Link State Routing Protocols*
- Internet draft draft-ietf-bfd-base-09.txt, *Bidirectional Forwarding Detection* (except for the transmission of echo packets)
- Internet draft draft-ietf-isis-wg-255adj-02.txt, *Maintaining more than 255 circuits in IS-IS*

To access Internet RFCs and drafts, go to the Internet Engineering Task Force (IETF) Web site at http://www.ietf.org.

Chapter 15

IS-IS Configuration Guidelines

This chapter discusses the following topics that provide information about configuring IS-IS:

- Configuring IS-IS on page 322
- Minimum IS-IS Configuration on page 324
- Configuring IS-IS Authentication on page 325
- Configuring of Interface-Specific IS-IS Properties on page 327
- Configuring BFD for IS-IS on page 328
- Overview of BFD Authentication for IS-IS on page 331
- Configuring BFD Authentication for IS-IS on page 333
- Enabling Packet Checksum on IS-IS Interfaces on page 336
- Configuring the Transmission Frequency for CSNP Packets on IS-IS Interfaces on page 336
- Configuring Synchronization Between LDP and IS-IS on page 336
- Configuring the Transmission Frequency for Link-State PDUs on IS-IS Interfaces on page 337
- Configuring Mesh Groups of IS-IS Interfaces on page 337
- Configuring IS-IS Multicast Topologies on page 338
- Configuring IS-IS IPv6 Unicast Topologies on page 340
- Configuring Point-to-Point Interfaces for IS-IS on page 340
- Configuring Levels on IS-IS Interfaces on page 341
- Configuring the Reference Bandwidth Used in IS-IS Metric Calculations on page 345
- Limiting the Number of Advertised IS-IS Areas on page 346
- Enabling Wide IS-IS Metrics for Traffic Engineering on page 346
- Configuring Preference Values for IS-IS Routes on page 347
- Limiting the Number of Prefixes Exported to IS-IS on page 347
- Configuring Link-State PDU Lifetime for IS-IS on page 347
- Advertising Label-Switched Paths into IS-IS on page 348
- Configuring IS-IS to Make Routing Devices Appear Overloaded on page 348
- Configuring SPF Options for IS-IS on page 349
- Configuring Graceful Restart for IS-IS on page 350

- Configuring IS-IS for Multipoint Network Clouds on page 351
- Configuring IS-IS Traffic Engineering Attributes on page 351
- Enabling Authentication for IS-IS Without Network-Wide Deployment on page 354
- Configuring Quicker Advertisement of IS-IS Adjacency State Changes on page 355
- Enabling Padding of IS-IS Hello Packets on page 355
- Configuring CLNS for IS-IS on page 356
- Disabling IS-IS on page 358
- Disabling IPv4 Routing for IS-IS on page 359
- Disabling IPv6 Routing for IS-IS on page 359
- Applying Policies to Routes Exported to IS-IS on page 360
- Installing a Default Route to the Nearest Routing Device That Operates at Both IS-IS Levels on page 362
- Configuring Loop-Free Alternate Routes for IS-IS on page 363
- Tracing IS-IS Protocol Traffic on page 369

Configuring IS-IS

To configure IS-IS, you include the following statements in the configuration:

```
protocols {
    isis {
        clns-routing;
        disable;
        ignore-attached-bit;
        graceful-restart {
            disable;
            helper-disable;
            restart-duration seconds;
        }
        label-switched-path name level level metric metric;
        level level-number {
            authentication-key key;
            authentication-type authentication;
            external-preference preference;
            no-csnp-authentication;
            no-hello-authentication;
            no-psnp-authentication;
            preference preference;
            prefix-export-limit number;
            wide-metrics-only;
        }
        loose-authentication-check;
        lsp-lifetime seconds;
        max-areas seconds;
        no-adjacency-holddown;
        no-authentication-check;
        no-ipv4-routing;
        no-ipv6-routing;
        overload {
```

```
    advertise-high-metrics;
    timeout seconds;
}
reference-bandwidth reference-bandwidth;
rib-group {
    inet group-name;
    inet6 group-name;
}
spf-options {
    delay milliseconds;
    holddown milliseconds;
    rapid-runs number;
}
topologies {
    ipv4-multicast;
    ipv6-multicast;
    ipv6-unicast;
}
traffic-engineering {
    disable;
    ignore-lsp-metrics;
    family inet;
        shortcuts {
            multicast-rpf-routes;
        }
    }
    family inet6;
        shortcuts;
    }
}
traceoptions {
    file filename <files number> <size size> <world-readable | no-world-readable>;
    flag flag <flag-modifier> <disable>;
}
interface (all | interface-name) {
    disable;
    bfd-liveness-detection {
        authentication {
            algorithm algorithm-name;
            key-chain key-chain-name;
            loose-check;
        }
        detection-time {
            threshold milliseconds;
        }
        minimum-interval milliseconds;
        minimum-receive-interval milliseconds;
        multiplier number;
        no-adaptation;
        transmit-interval {
            threshold milliseconds;
            minimum-interval milliseconds;
        }
        version (1 | automatic);
    }
    checksum;
```

```
            csnp-interval (seconds | disable);
            hello-padding (adaptive | loose | strict);
            ldp-synchronization {
               disable;
               hold-time seconds;
            }
            link-protection;
            lsp-interval milliseconds;
            mesh-group (value | blocked);
            no-adjacency-holddown;
            no-eligible-backup;
            no-ipv4-multicast;
            no-ipv6-multicast;
            no-ipv6-unicast;
            no-unicast-topology;
            node-link-protection;
            passive;
            point-to-point;
            level level-number {
               disable;
               hello-authentication-key key;
               hello-authentication-type authentication;
               hello-interval seconds;
               hold-time seconds;
               ipv4-multicast-metric number;
               ipv6-multicast-metric number;
               ipv6-unicast-metric number;
               metric metric;
               passive;
               priority number;
               te-metric metric;
            }
         }
      }
   }
```

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for these statements.

By default, IS-IS is enabled for Level 1 and Level 2 routers on all interfaces on which an International Organization for Standardization (ISO) address is configured.

Minimum IS-IS Configuration

For IS-IS to run on the routing device, you must enable IS-IS on the routing device, configure a network entity title (NET) on one of the routing device's interfaces (preferably the loopback interface, lo0), and configure the ISO family on all interfaces on which you want IS-IS to run. When you enable IS-IS, Level 1 and Level 2 are enabled by default. The following is the minimum IS-IS configuration. In the address statement, *address* is the NET:

```
interfaces {
   lo0 {
      unit logical-unit-number {
```

```
            family iso {
               address address;
            }
         }
      }
      interface-type-fpc/pic/port {
         unit logical-unit-number {
            family iso;
         }
      }
   }
   protocols {
      isis {
         interface all;
      }
   }
```

NOTE: To create the IS-IS interface, you must also configure IS-IS at the [edit protocols isis interface *interface-name*] hierarchy level. If you want the JUNOS Software to create IS-IS interfaces automatically, include the interface all option at the [edit protocols isis] hierarchy level.

Configuring IS-IS Authentication

All IS-IS protocol exchanges can be authenticated to guarantee that only trusted routing devices participate in the autonomous system (AS) routing. By default, IS-IS authentication is disabled on the routing device.

To configure IS-IS authentication, you must define an authentication password and specify the authentication type.

You can configure one of the following authentication methods:

- Simple authentication—Uses a text password that is included in the transmitted packet. The receiving routing device uses an authentication key (password) to verify the packet. Simple authentication is included for compatibility with existing IS-IS implementations. However, we recommend that you do *not* use this authentication method because it is insecure (the text can be "sniffed").

CAUTION: A simple password that exceeds 254 characters is truncated.

- HMAC-MD5 authentication—Uses an iterated cryptographic hash function. The receiving routing device uses an authentication key (password) to verify the packet.

You can also configure more fine-grained authentication for hello packets. To do this, see "Configuring Authentication for IS-IS Hello Packets" on page 343.

To enable authentication and specify an authentication method, include the authentication-type statement, specifying the simple or md5 authentication type:

```
authentication-type authentication;
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

To configure a password, include the authentication-key statement. The authentication password for all routing devices in a domain must be the same.

```
authentication-key key;
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

The password can contain up to 255 characters. If you include spaces, enclose all characters in quotation marks (“ ”).

If you are using the JUNOS IS-IS software with another implementation of IS-IS, the other implementation must be configured to use the same password for the domain, the area, and all interfaces that are shared with a JUNOS implementation.

Authentication of hello packets, partial sequence number PDU (PSNP), and complete sequence number PDU (CSNP) may be suppressed to enable interoperability with the routing software of different vendors. Different vendors handle authentication in various ways, and suppressing authentication for different PDU types may be the simplest way to allow compatibility within the same network.

To configure IS-IS to generate authenticated packets, but not to check the authentication on received packets, include the no-authentication-check statement:

```
no-authentication-check;
```

To suppress authentication of IS-IS hello packets, include the no-hello-authentication statement:

```
no-hello-authentication;
```

To suppress authentication of PSNP packets, include the no-psnp-authentication statement:

```
no-psnp-authentication;
```

To suppress authentication of CSNP packets, include the no-csnp-authentication statement:

```
no-csnp-authentication;
```

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for these statements.

NOTE: The authentication and the no-authentication statements must be configured at the same hierarchy level. Configuring authentication at the interface hierarchy level and configuring no-authentication at the isis hierarchy level has no effect.

Configuring of Interface-Specific IS-IS Properties

You can configure interface-specific IS-IS properties by including the interface statement.

```
interface (all | interface-name) {
    disable;
    bfd-liveness-detection {
        authentication {
            algorithm algorithm-name;
            key-chain key-chain-name;
            loose-check;
        }
        detection-time {
            threshold milliseconds;
        }
        minimum-interval milliseconds;
        minimum-receive-interval milliseconds;
        transmit-interval {
            threshold milliseconds;
            minimum-interval milliseconds;
        }
        multiplier number;
        version (1 | automatic);
    }
    checksum;
    csnp-interval (seconds | disable);
    ldp-synchronization {
        disable;
        hold-time seconds;
    }
    lsp-interval milliseconds;
    mesh-group (value | blocked);
    no-ipv4-multicast;
    no-ipv6-multicast;
    no-ipv6-unicast;
    no-unicast-topology;
    passive;
    point-to-point;
    level level-number {
        disable;
        hello-authentication-type authentication;
        hello-authentication-key key;
        hello-interval seconds;
        hold-time seconds;
        ipv4-multicast-metric number;
        ipv6-multicast-metric number;
        ipv6-unicast-metric number;
```

```
        metric metric;
        passive;
        priority number;
        te-metric metric;
    }
}
```

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for these statements.

For *interface-name*, specify the full interface name, including the physical and logical address components. To configure all interfaces, specify the interface name as **all**. For information about configuring interfaces, see the *JUNOS Network Interfaces Configuration Guide*.

For more information about configuring IS-IS interface properties, see the following topics:

- Configuring BFD for IS-IS on page 328
- Overview of BFD Authentication for IS-IS on page 331
- Configuring BFD Authentication for IS-IS on page 333
- Enabling Packet Checksum on IS-IS Interfaces on page 336
- Configuring the Transmission Frequency for CSNP Packets on IS-IS Interfaces on page 336
- Configuring Synchronization Between LDP and IS-IS on page 336
- Configuring the Transmission Frequency for Link-State PDUs on IS-IS Interfaces on page 337
- Configuring Mesh Groups of IS-IS Interfaces on page 337
- Configuring IS-IS Multicast Topologies on page 338
- Configuring IS-IS IPv6 Unicast Topologies on page 340
- Configuring Point-to-Point Interfaces for IS-IS on page 340
- Configuring Levels on IS-IS Interfaces on page 341

Configuring BFD for IS-IS

The Bidirectional Forwarding Detection (BFD) protocol is a simple hello mechanism that detects failures in a network. Hello packets are sent at a specified, regular interval. A neighbor failure is detected when the routing device stops receiving a reply after a specified interval. BFD works with a wide variety of network environments and topologies. The failure detection timers for BFD have shorter time limits than the failure detection mechanisms of IS-IS, providing faster detection. These timers are also adaptive and can be adjusted to be more or less aggressive. For example, the timers can adapt to a higher value if the adjacency fails, or a neighbor can negotiate a higher value for a timer than the configured.

NOTE: BFD for IS-IS on an IPv6-only interface only is not supported. However, if the interface is dual-stacked (both IPV4 and IPV6 are configured) then you can configure BFD as a client on the IPv4 IS-IS session."

To enable failure detection, include the bfd-liveness-detection statement:

```
bfd-liveness-detection {
    detection-time {
        threshold milliseconds;
    }
    minimum-interval milliseconds;
    minimum-receive-interval milliseconds;
    no-adaptation;
    transmit-interval {
        threshold milliseconds;
        minimum-interval milliseconds;
    }
    multiplier number;
    version (1 | automatic);
}
```

To specify the threshold for the adaptation of the detection time, include the threshold statement:

```
detection-time {
    threshold milliseconds;
}
```

When the BFD session detection time adapts to a value equal to or higher than the threshold, a single trap and a system log message are sent.

To specify the minimum transmit and receive intervals for failure detection, include the minimum-interval statement:

```
minimum-interval milliseconds;
```

This value represents the minimum interval at which the local router transmits hellos packets as well as the minimum interval at which the router expects to receive a reply from a neighbor with which it has established a BFD session. You can configure a number in the range from 1 through 255,000 milliseconds. You can also specify the minimum transmit and receive intervals separately

NOTE: Specifying an interval less than 300 milliseconds can cause undesired BFD flapping.

To specify only the minimum receive interval for failure detection, include the minimum-receive-interval statement:

```
minimum-receive-interval milliseconds;
```

This value represents the minimum interval at which the local router expects to receive a reply from a neighbor with which it has established a BFD session. You can configure a number in the range of 1 through 255,000 milliseconds.

To specify the number of hello packets not received by the neighbor that causes the originating interface to be declared down, include the multiplier statement:

```
multiplier number;
```

The default is 3, and you can configure a value in the range from 1 through 225.

To specify the minimum transmit interval for failure detection, include the transmit-interval minimum-interval statement:

```
transmit-interval {
    minimum-interval milliseconds;
}
```

This value represents the minimum interval at which the local routing device transmits hello packets to the neighbor with which it has established a BFD session. You can configure a value in the range from 1 through 255,000 milliseconds.

To specify the threshold for detecting the adaptation of the transmit interval, include the threshold statement:

```
transmit-interval {
    threshold milliseconds;
}
```

The threshold value must be greater than the minimum transmit interval.

You can trace BFD operations by including the traceoptions statement at the [edit protocols bfd] hierarchy level. For more information, see "Tracing BFD Protocol Traffic" on page 78.

In JUNOS Release 9.0 and later, you can specify that the BFD sessions not adapt to changing network conditions. To disable BFD adaptation, include the no-adaptation statement:

```
no-adaptation;
```

NOTE: We recommend that you not disable BFD adaptation unless it is preferable not to have BFD adaptation enabled in your network.

To specify the BFD version used for detection, include the version statement:

```
version (1 | automatic);
```

The default is to have the version detected automatically.

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for these statements.

Overview of BFD Authentication for IS-IS

BFD enables rapid detection of communication failures between adjacent systems. By default, authentication for BFD sessions is disabled. However, when running BFD over Network Layer protocols, the risk of service attacks can be significant. We strongly recommend using authentication if you are running BFD over multiple hops or through insecure tunnels. Beginning with JUNOS Release 9.6, the JUNOS Software supports authentication for BFD sessions running over IS-IS. BFD authentication is only supported in the domestic image and is not available in the export image.

You authenticate BFD sessions by specifying an authentication algorithm and keychain, and then associating that configuration information with a security authentication keychain using the keychain name.

The following sections describe the supported authentication algorithms, security keychains, and level of authentication that can be configured:

- BFD Authentication Algorithms on page 331
- Security Authentication Keychains on page 332
- Strict Versus Loose Authentication on page 332

BFD Authentication Algorithms

JUNOS Software supports the following algorithms for BFD authentication:

- simple-password—Plain-text password. One to 16 bytes of plain text are used to authenticate the BFD session. One or more passwords may be configured. This method is the least secure and should be used only when BFD sessions are not subject to packet interception.
- keyed-md5—Keyed Message Digest 5 hash algorithm for sessions with transmit and receive intervals greater than 100 ms. To authenticate the BFD session, keyed MD5 uses one or more secret keys (generated by the algorithm) and a sequence number that is updated periodically. With this method, packets are accepted at the receiving end of the session if one of the keys matches and the sequence number is greater than or equal to the last sequence number received. Although more secure than a simple password, this method is vulnerable to replay attacks. Increasing the rate at which the sequence number is updated can reduce this risk.
- meticulous-keyed-md5—Meticulous keyed Message Digest 5 hash algorithm. This method works in the same manner as keyed MD5, but the sequence number is updated with every packet. Although more secure than keyed MD5 and simple passwords, this method may take additional time to authenticate the session.
- keyed-sha-1—Keyed Secure Hash Algorithm I for sessions with transmit and receive intervals greater than 100 ms. To authenticate the BFD session, keyed SHA uses one or more secret keys (generated by the algorithm) and a sequence number that is updated periodically. The key is not carried within the packets. With this method, packets are accepted at the receiving end of the session if one

of the keys matches and the sequence number is greater than the last sequence number received.

- meticulous-keyed-sha-1—Meticulous keyed Secure Hash Algorithm I. This method works in the same manner as keyed SHA, but the sequence number is updated with every packet. Although more secure than keyed SHA and simple passwords, this method may take additional time to authenticate the session.

NOTE: Nonstop active routing (NSR) is not supported with meticulous-keyed-md5 and meticulous-keyed-sha-1 authentication algorithms. BFD sessions using these algortihms may go down after a switchover.

Security Authentication Keychains

The security authentication keychain defines the authentication attributes used for authentication key updates. When the security authentication keychain is configured and associated with a protocol through the keychain name, authentication key updates can occur without interrupting routing and signaling protocols.

The authentication keychain contains one or more keychains. Each keychain contains one or more keys. Each key holds the secret data and the time at which the key becomes valid. The algorithm and keychain must be configured on both ends of the BFD session, and they must match. Any mismatch in configuration prevents the BFD session from being created.

BFD allows multiple clients per session, and each client can have its own keychain and algorithm defined. To avoid confusion, we recommend specifying only one security authentication keychain.

Strict Versus Loose Authentication

By default, strict authentication is enabled and authentication is checked at both ends of each BFD session. Optionally, to smooth migration from nonauthenticated sessions to authenticated sessions, you can configure *loose checking*. When loose checking is configured, packets are accepted without authentication being checked at each end of the session. This feature is intended for transitional periods only.

Related Topics

- Configuring BFD Authentication for IS-IS on page 333
- bfd-liveness-detection statement
- authentication-key-chains statement in the *JUNOS System Basics Configuration Guide*
- show bfd session command in the *JUNOS Routing Protocols and Policies Command Reference*
- Configuring BFD for IS-IS on page 328

Configuring BFD Authentication for IS-IS

Beginning with JUNOS Release 9.6, you can configure authentication for BFD sessions running over IS-IS. Routing instances are also supported. Only three steps are needed to configure authentication on a BFD session:

1. Specify the BFD authentication algorithm for the IS-IS protocol.
2. Associate the authentication keychain with the IS-IS protocol.
3. Configure the related security authentication keychain.

The following sections provide instructions for configuring and viewing BFD authentication on IS-IS:

- Configuring BFD Authentication Parameters on page 333
- Viewing Authentication Information for BFD Sessions on page 334

Configuring BFD Authentication Parameters

To configure BFD authentication:

1. Specify the algorithm (keyed-md5, keyed-sha-1, meticulous-keyed-md5, meticulous-keyed-sha-1, or simple-password) to use for BFD authentication on an IS-IS route or routing instance.

   ```
   [edit]
   user@host# set protocols isis interface if1-isis bfd-liveness-detection
     authentication algorithm keyed-sha-1
   ```

NOTE: Nonstop active routing (NSR) is not supported with meticulous-keyed-md5 and meticulous-keyed-sha-1 authentication algorithms. BFD sessions using these algortihms may go down after a switchover.

2. Specify the keychain to be used to associate BFD sessions on the specified IS-IS route or routing instance with the unique security authentication keychain attributes. This should match the keychain name configured at the [edit security authentication key-chains] hierarchy level.

   ```
   [edit]
   user@host# set protocols isis interface if1-isis bfd-liveness-detection
     authentication keychain bfd-isis
   ```

NOTE: The algorithm and keychain must be configured on both ends of the BFD session, and they must match. Any mismatch in configuration prevents the BFD session from being created.

3. Specify the unique security authentication information for BFD sessions:

- The matching *key-chain-name* as specified in step 2.
- At least one *key*, a unique integer between 0 and 63. Creating multiple keys allows multiple clients to use the BFD session.
- The *secret-data* used to allow access to the session.
- The time at which the authentication key becomes active, *yyyy-mm-dd.hh:mm:ss*.

```
[edit security]
user@host# authentication-key-chains key-chain bfd-sr4 key 53 secret
   $9$ggaJDmPQ6/tJgF/AtREVsyPsnCtUHm start-time 2009-06-14.10:00:00
```

4. (Optional) Specify loose authentication checking if you are transitioning from nonauthenticated sessions to authenticated sessions.

```
[edit]
user@host> set protocols isis interface if1-isis bfd-liveness-detection
   authentication loose-check
```

5. (Optional) View your configuration using the show bfd session detail or show bfd session extensive command.
6. Repeat these steps to configure the other end of the BFD session.

NOTE: BFD authentication is only supported in the domestic image and is not available in the export image.

Viewing Authentication Information for BFD Sessions

You can view the existing BFD authentication configuration using the show bfd session detail and show bfd session extensive commands.

The following example shows BFD authentication configured for the if1-isis interface. It specifies the keyed SHA-1 authentication algorithm and a keychain name of bfd-isis. The authentication keychain is configured with two keys. Key 1 contains the secret data "9ggaJDmPQ6/tJgF/AtREVsyPsnCtUHm" and a start time of June 1, 2009, at 9:46:02 AM PST. Key 2 contains the secret data "9a5jiKW9I.reP38ny.TszF2/9" and a start time of June 1, 2009, at 3:29:20 PM PST.

```
[edit protocols isis]
interface if1-isis {
   bfd-liveness-detection {
      authentication {
         algorithm keyed-sha-1;
         key-chain bfd-isis;
      }
   }
}
[edit security]
authentication key-chains {
```

```
    key-chain bfd-isis {
      key 1 {
        secret "$9$ggaJDmPQ6/tJgF/AtREVsyPsnCtUHm";
        start-time "2009-6-1.09:46:02 -0700";
      }
      key 2 {
        secret "$9$a5jiKW9I.reP38ny.TszF2/9";
        start-time "2009-6-1.15:29:20 -0700";
      }
    }
  }
```

If you commit these updates to your configuration, you would see output similar to the following. In the output for the show bfd sessions detail command, Authenticate is displayed to indicate that BFD authentication is configured. For more information about the configuration, use the show bfd sessions extensive command. The output for this command provides the keychain name, the authentication algorithm and mode for each client in the session, and the overall BFD authentication configuration status, keychain name, and authentication algorithm and mode.

show bfd sessions detail

```
user@host# show bfd session detail

                                                  Detect   Transmit
Address                  State     Interface      Time     Interval  Multiplier
10.9.1.29                 Up        ge-4/0/0.0     0.600     0.200        3
 Client ISIS L2, TX interval 0.200, RX interval 0.200, multiplier 3, Authenticate

 Session up time 3d 00:34, previous down time 00:00:01
 Local diagnostic NbrSignal, remote diagnostic AdminDown
 Remote state Up, version 1

1 sessions, 1 clients
Cumulative transmit rate 10.0 pps, cumulative receive rate 10.0 pps
```

show bfd sessions extensive

```
user@host# show bfd session extensive
                                                  Detect   Transmit
Address                  State     Interface      Time     Interval  Multiplier
10.9.1.29            Up        ge-4/0/0.0          0.600     0.200      3
  Client ISIS L2, TX interval 0.200, RX interval 0.200, multiplier 3, Authenticate

        keychain bfd-isis, algo keyed-sha-1, mode strict
  Session up time 00:04:42
 Local diagnostic None, remote diagnostic NbrSignal
 Remote state Up, version 1
 Replicated
 Min async interval 0.300, min slow interval 1.000
 Adaptive async TX interval 0.300, RX interval 0.300
 Local min TX interval 0.300, minimum RX interval 0.300, multiplier 3
 Remote min TX interval 0.300, min RX interval 0.300, multiplier 3
 Local discriminator 2, remote discriminator 2
 Echo mode disabled/inactive
 Authentication enabled/active, keychain bfd-isis, algo keyed-sha-1, mode strict


1 sessions, 1 clients

Cumulative transmit rate 10.0 pps, cumulative receive rate 10.0 pps
```

Related Topics

- Overview of BFD Authentication for IS-IS on page 331
- bfd-liveness-detection statement
- authentication-key-chains statement in the *JUNOS System Basics Configuration Guide*
- show bfd session command in the *JUNOS Routing Protocols and Policies Command Reference*
- Configuring BFD for IS-IS on page 328

Enabling Packet Checksum on IS-IS Interfaces

You can enable checksum for packets on a per-interface basis. To enable checksum, include the checksum statement:

```
checksum;
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

Configuring the Transmission Frequency for CSNP Packets on IS-IS Interfaces

By default, IS-IS sends complete sequence number (CSN) packets periodically. If the routing device is the designated router on a LAN, IS-IS sends CSN packets every 10 seconds. If the routing device is on a point-to-point interface, it sends CSN packets every 5 seconds. You might want to modify the default interval to protect against link-state PDU (LSP) flooding.

To modify the CSNP interval, include the csnp-interval statement:

```
csnp-interval seconds;
```

The time can range from 1 through 65,535 seconds.

To configure the interface not to send any CSN packets, specify the disable option:

```
csnp-interval disable;
```

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for these statements.

Configuring Synchronization Between LDP and IS-IS

LDP distributes labels in non-traffic-engineered applications. Labels are distributed along the best path determined by IS-IS. If the synchronization between LDP and IS-IS is lost, the label-switched path (LSP) goes down. Therefore, LDP and IS-IS synchronization is beneficial. When LDP is not fully operational on a given link (a session is not established and labels are not exchanged), IS-IS advertises the link with the maximum cost metric. The link is not preferred but remains in the network topology.

LDP synchronization is supported only on point-to-point interfaces and LAN interfaces configured as point-to-point interfaces under IS-IS. LDP synchronization is not supported during graceful restart.

To advertise the maximum cost metric until LDP is operational for LDP synchronization, include the ldp-synchronization statement:

```
ldp-synchronization {
   disable;
   hold-time seconds;
}
```

To disable synchronization, include the disable statement. To configure the time period to advertise the maximum cost metric for a link that is not fully operational, include the hold-time statement.

NOTE: When an interface has been in the holddown state for more than three minutes, a system log message with a warning level is sent. This message appears in both the messages file and the trace file.

For a list of hierarchy levels at which you can include these statements, see the statement summary section for these statements.

Configuring the Transmission Frequency for Link-State PDUs on IS-IS Interfaces

By default, the routing device sends one link-state PDU packet out an interface every 100 milliseconds. To modify this interval, include the lsp-interval statement:

```
lsp-interval milliseconds;
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

To disable the transmission of all link-state PDU packets, set the interval to 0.

Configuring Mesh Groups of IS-IS Interfaces

A *mesh group* is a set of routing devices that are fully connected; that is, they have a fully meshed topology. When link-state PDU packets are being flooded throughout an area, each router within a mesh group receives only a single copy of an link-state PDU packet instead of receiving one copy from each neighbor, thus minimizing the overhead associated with the flooding of link-state PDU packets.

To create a mesh group and designate that an interface is part of the group, assign a mesh-group number to all the routing device interfaces in the group:

```
mesh-group value;
```

To prevent an interface in the mesh group from flooding link-state PDUs, configure blocking on that interface:

```
mesh-group blocked;
```

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for these statements.

Configuring IS-IS Multicast Topologies

Most multicast routing protocols perform a reverse-path forwarding (RPF) check on the source of multicast data packets. If a packet comes in on the interface that is used to send data to the source, the packet is accepted and forwarded to one or more downstream interfaces. Otherwise, the packet is discarded and a notification is sent to the multicast routing protocol running on the interface.

In certain instances, the unicast routing table used for the RPF check is also the table used for forwarding unicast data packets. Thus, unicast and multicast routing are congruent. In other cases, where it is preferred that multicast routing be independent of unicast routing, the multicast routing protocols are configured to perform the RPF check using an alternate unicast routing table inet.2.

You can configure IS-IS to calculate an alternate IPv4 multicast topology, in addition to the normal IPv4 unicast topology, and add the corresponding routes to inet.2. The IS-IS interface metrics for the multicast topology can be configured independently of the unicast metrics. You can also selectively disable interfaces from participating in the multicast topology while continuing to participate in the regular unicast topology. This lets you exercise control over the paths that multicast data takes through a network so that it is independent of unicast data paths.

You can also configure IS-IS to calculate an alternate IPv6 multicast topology, in addition to the normal IPv6 unicast topology.

To enable an alternate IPv4 multicast topology for IS-IS, include the ipv4-multicast statement:

```
ipv4-multicast;
```

To configure the multicast metric for an alternate multicast topology, include the ipv4-multicast-metric statement:

```
ipv4-multicast-metric number;
```

To exclude an interface from the multicast topology for IS-IS, include the no-ipv4-multicast statement:

```
no-ipv4-multicast;
```

To enable an alternate IPv6 multicast topology for IS-IS, include the ipv6-multicast statement:

```
ipv6-multicast;
```

To configure the multicast metric for an alternate IPv6 multicast topology, include the ipv6-multicast-metric statement:

```
ipv6-multicast-metric number;
```

To exclude an interface from the IPv6 multicast topology for IS-IS, include the no-ipv6-multicast statement:

```
no-ipv6-multicast;
```

To exclude an interface from the IPv4 unicast topologies for IS-IS, include the no-unicast-topology statement:

```
no-unicast-topology;
```

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for these statements.

Example: Configuring IS-IS Multicast Topologies

```
[edit]
protocols {
    isis {
        traceoptions {
            file isis size 5m world-readable;
        }
        isis so-0/0/0.0 {
            level 1 {
                metric 15;
                multicast-metric 18;
            }
            level 2 {
                metric 20;
                multicast-metric 14;
            }
        }
        isis so-1/0/0.0 {
            level 1 {
                metric 15;
                multicast-metric 12;
            }
            level 2 {
                metric 20;
                multicast-metric 23;
            }
        }
        isis so-2/0/0.0 {
            no-multicast;
            level 1 metric 14;
            level 2 metric 23;
        }
        isis fxp0.0 {
            disable;
        }
    }
}
```

Configuring IS-IS IPv6 Unicast Topologies

You can configure IS-IS to calculate an alternate IPv6 unicast topology, in addition to the normal IPv4 unicast topology, and add the corresponding routes to inet6.0. The IS-IS interface metrics for the IPv4 topology can be configured independently of the IPv6 metrics. You can also selectively disable interfaces from participating in the IPv6 topology while continuing to participate in the IPv4 topology. This lets you exercise control over the paths that unicast data takes through a network.

To enable an alternate IPv6 unicast topology for IS-IS, include the ipv6-unicast statement:

```
isis {
    topologies {
        ipv6-unicast;
    }
}
```

To configure a metric for an alternate IPv6 unicast topology, include the ipv6-unicast-metric statement:

```
isis {
    interface interface-name {
        level level-number {
            ipv6-unicast-metric number;
        }
    }
}
```

To exclude an interface from the IPv6 unicast topologies for IS-IS, include the no-ipv6-unicast statement:

```
isis {
    interface interface-name {
        no-ipv6-unicast;
    }
}
```

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for these statements.

Configuring Point-to-Point Interfaces for IS-IS

You can use the point-to-point statement to configure a LAN interface to act like a point-to-point interface for IS-IS. You do not need an unnumbered LAN interface, and it has no effect if configured on an interface that is already point-to-point.

The point-to-point statement affects only IS-IS protocol procedures on that interface; all other protocols continue to treat the interface as a LAN interface. Only two IS-IS routing devices can be connected to the LAN interface and both must be configured as point-to-point.

To configure a point-to-point IS-IS interface, include the **point-to-point** statement:

```
point-to-point;
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

Configuring Levels on IS-IS Interfaces

You can administratively divide a single AS into smaller groups called areas. You configure each routing device interface to be in an area. Any interface can be in any area. The area address applies to the entire routing device; you cannot specify one interface to be in one area and another interface in a different area. In order to route between areas you must have two adjacent Level 2 routers that communicate with each other.

Level 1 routers can only route within their IS-IS area. To send traffic outside their area, Level 1 routers must send packets to the nearest intra-area Level 2 router. A routing device can be a Level 1 router, a Level 2 router, or both. You specify the router level on a per-interface basis, and a routing device becomes adjacent with other routing devices on the same level on that link only.

You can configure one Level 1 routing process and one Level 2 routing process on each interface, and you can configure the two levels differently.

To configure an area, include the **level** statement:

```
level level-number {
    disable;
    hello-authentication-key key;
    hello-authentication-type authentication;
    hello-interval seconds;
    hold-time seconds;
    ipv4-multicast-metric number;
    ipv6-multicast-metric number;
    ipv6-unicast-metric number;
    metric metric;
    passive;
    priority number;
    te-metric metric;
}
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

The statements within the **level** statement allow you to perform the following tasks when configuring the following optional level-specific properties:

- Disabling IS-IS at a Level on IS-IS Interfaces on page 342
- Advertising Interface Addresses Without Running IS-IS on page 343
- Configuring Authentication for IS-IS Hello Packets on page 343
- Configuring the Transmission Frequency for IS-IS Hello Packets on page 343

- Configuring the Delay Before IS-IS Neighbors Mark the Routing Device as Down on page 344
- Configuring the Metric Value for IS-IS Routes on page 344
- Configuring the IS-IS Metric Value Used for Traffic Engineering on page 344
- Configuring the Designated Router Priority for IS-IS on page 345
- Advertising Interface Addresses Without Running IS-IS on page 345

Disabling IS-IS at a Level on IS-IS Interfaces

By default, IS-IS is enabled for IS-IS areas on all enabled interfaces on which the ISO protocol family is enabled (at the [edit interfaces *interface* unit *logical-unit-number*] hierarchy level). To disable IS-IS at any particular level on an interface, include the disable statement:

```
disable;
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

Enabling IS-IS on an interface (by including the interface statement at the [edit protocols isis] hierarchy level), disabling it (by including the disable statement), and not actually having IS-IS run on an interface (by including the passive statement) are mutually exclusive states.

Example: Disabling IS-IS at a Level

On SONET/SDH interface so-0/0/0, enable IS-IS for Level 1 only. With this configuration, tracing messages periodically indicate that IS-IS is creating Level 2 link-state PDUs. However, because IS-IS for Level 2 is disabled, these link-state PDUs are never distributed to neighboring routers.

```
protocols {
    isis {
        traceoptions {
            file isis size 1m files 10;
            flag spf;
            flag lsp;
            flag error;
        }
        interface so-0/0/0 {
            level 2 {
                disable;
            }
        }
    }
}
```

Advertising Interface Addresses Without Running IS-IS

By default, IS-IS must be configured on an interface or a level for direct interface addresses to be advertised into that level. To advertise the direct interface addresses without actually running IS-IS on that interface or level, include the passive statement:

```
passive;
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

Enabling IS-IS on an interface (by including the interface statement at the [edit protocols isis] hierarchy level), disabling it (by including the interface disable statement), and not actually having IS-IS run on an interface (by including the passive statement) are mutually exclusive states.

NOTE: If neither passive mode nor family ISO are configured on the IS-IS interface, then the routing device treats the interface as not being operational and no direct IPv4/IPv6 routes are exported into IS-IS.

Configuring Authentication for IS-IS Hello Packets

You can configure authentication configure authentication for a given IS-IS level on an interface. On a point-to-point link, if you enable hello authentication for both IS-IS levels, the password configured for Level 1 is used for both levels.

CAUTION: If no authentication is configured for Level 1 on a point-to-point link with both levels enabled, the hello packets are sent without any password, regardless of the Level 2 authentication configurations.

By default, hello authentication is not configured on an interface. However, if IS-IS authentication is configured, the hello packets are authenticated using the IS-IS authentication type and password.

To enable hello authentication for an IS-IS level on an interface and define the password, include the hello-authentication-type and hello-authentication-key statements:

```
hello-authentication-type (md5 | simple);
hello-authentication-key password;
```

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for these statements.

Configuring the Transmission Frequency for IS-IS Hello Packets

Routing devices send hello packets at a fixed interval on all interfaces to establish and maintain neighbor relationships. This interval is advertised in the hello interval

field in the hello packet. By default, a designated intersystem (DIS) router sends hello packets every 3 seconds, and a non-DIS router sends hello packets every 9 seconds.

To modify how often the routing device sends hello packets out of an interface, include the hello-interval statement:

hello-interval *seconds*;

The hello interval range is from 1 through 20,000 seconds.

You can send out hello packets in sub-second intervals. To send out hello packets every 333 milliseconds, set the hello-interval value to 1.

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

Configuring the Delay Before IS-IS Neighbors Mark the Routing Device as Down

The hold time specifies how long a neighbor should consider this routing device to be operative without receiving another hello packet. If the neighbor does not receive a hello packet from this routing device within the hold time, it marks the routing device as being unavailable. The default hold-time value is three times the default hello interval: 9 seconds for a DIS router and 27 seconds for a non-DIS router.

To modify the hold-time value on the local routing device, include the hold-time statement:

hold-time *seconds*;

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

Configuring the Metric Value for IS-IS Routes

All IS-IS routes have a cost, which is a routing metric that is used in the IS-IS link-state calculation. The cost is an arbitrary, dimensionless integer that can be from 1 through 63, or from 1 through 16,777,215 ($2^{24} - 1$) if you are using wide metrics. The default metric value is 10 (with the exception of the lo0 interface, which has a default metric of 0). To modify the default value, include the metric statement:

metric *metric*;

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

For more information about IS-IS interface metrics, see “Configuring the Reference Bandwidth Used in IS-IS Metric Calculations” on page 345.

Configuring the IS-IS Metric Value Used for Traffic Engineering

When traffic engineering is enabled on the routing device, you can configure an IS-IS metric that is used exclusively for traffic engineering. The traffic engineering metric

is used for information injected into the traffic engineering database. Its value does not affect normal IS-IS forwarding.

To modify the default value, include the te-metric statement:

```
te-metric metric;
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

Configuring the Designated Router Priority for IS-IS

A routing device advertises its priority to become a designated router in its hello packets. On all multiaccess networks, IS-IS uses the advertised priorities to elect a designated router for the network. This routing device is responsible for sending network link-state advertisements, which describe all the routing devices attached to the network. These advertisements are flooded throughout a single area.

The priority value is meaningful only on a multiaccess network. It has no meaning on a point-to-point interface.

A routing device's priority for becoming the designated router is indicated by an arbitrary number from 0 through 127; routing devices with a higher value are more likely to become the designated router. By default, routing devices have a priority value of 64.

To modify the interface's priority value, include the priority statement:

```
priority number;
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

Advertising Interface Addresses Without Running IS-IS

The routing device can advertise the direct interface addresses on an interface or on a sub-level of the interface without actually running IS-IS on that interface or at that level. This occurs in passive mode.

To enable an interface as passive, include the passive statement:

```
passive;
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

Configuring the Reference Bandwidth Used in IS-IS Metric Calculations

All IS-IS interfaces have a cost, which is a routing metric that is used in the IS-IS link-state calculation. Routes with lower total path metrics are preferred over those with higher path metrics. When there are several equal-cost routes to a destination, traffic is distributed equally among them.

The cost of a route is described by a single dimensionless metric that is determined using the following formula:

```
cost = reference-bandwidth/bandwidth
```

To modify the reference bandwidth, include the reference-bandwidth statement:

```
reference-bandwidth reference-bandwidth;
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

reference-bandwidth is the reference bandwidth. If the reference bandwidth is not configured, all interfaces have a default metric of 10 (with the exception of the lo0 interface, which has a default metric of 0).

For example, if you set the reference bandwidth to 1 Gbps (that is, *reference-bandwidth* is set to 1,000,000,000), a 100-Mbps interface has a default metric of 10.

For more information about IS-IS route metrics, see "Configuring the Metric Value for IS-IS Routes" on page 344.

Limiting the Number of Advertised IS-IS Areas

By default, IS-IS advertises a maximum of three areas in the IS-IS hello (IIH) PDUs and link-state PDUs. To advertise more than three ISO network addresses for a router, include the max-areas statement:

```
max-areas number;
```

The range that you can configure is from 3 through 36, and the default is 3. This value is included in the Maximum Address Area field of the IS-IS common PDU header included in all outgoing PDUs.

NOTE: The maximum number areas you can advertise is restricted to 36 to ensure that the IIH PDUs have enough space to include other type, length, and value (TLV) fields, such as the Authentication and IPv4 and IPv6 Interface Address TLVs.

For a list of hierarchy levels at which you an configure this statement, see the statement summary section for this statement.

Enabling Wide IS-IS Metrics for Traffic Engineering

Normally, IS-IS metrics can have values up to 63, and IS-IS generates two type length values (TLVs), one for an IS-IS adjacency and the second for an IP prefix. To allow IS-IS to support traffic engineering, a second pair of TLVs has been added to IS-IS, one for IP prefixes and the second for IS-IS adjacency and traffic engineering information. With these TLVs, IS-IS metrics can have values up to 16,777,215 ($2^{24} - 1$).

By default, the JUNOS Software supports the sending and receiving of wide metrics. The JUNOS Software allows a maximum metric value of 63 and generates both pairs of TLVs. To configure IS-IS to generate only the new pair of TLVs and thus to allow the wider range of metric values, include the wide-metrics-only statement:

```
wide-metrics-only;
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

Configuring Preference Values for IS-IS Routes

Route preferences are used to select which route is installed in the forwarding table when several protocols calculate routes to the same destination. The route with the lowest preference value is selected. For more information about route preferences, see "Route Preferences Overview" on page 6.

By default, Level 1 IS-IS internal routes have a preference value of 15, Level 2 IS-IS internal routes have a preference of 18, Level 1 IS-IS external routes have a preference of 160, and Level 2 external routes have a preference of 165. To change the preference values, include the preference statement (for internal routes) or the external-preference statement:

```
external-preference preference;
preference preference;
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

The preference value can range from 0 through 4,294,967,295 ($2^{32} - 1$).

Limiting the Number of Prefixes Exported to IS-IS

By default, there is no limit to the number of prefixes that can be exported into IS-IS. To configure a limit to the number of prefixes that can be exported into IS-IS, include the prefix-export-limit statement:

```
prefix-export-limit number;
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

You can specify a number range from 0 through 4,294,967,295.

Configuring Link-State PDU Lifetime for IS-IS

By default, link-state PDUs are maintained in network databases for 1200 seconds (20 minutes) before being considered invalid. This length of time, called the *LSP lifetime*, normally is sufficient to guarantee that link-state PDUs never expire.

To modify the link-state PDU lifetime, include the lsp-lifetime statement:

lsp-lifetime *seconds*;

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

The time can range from 350 through 65,535 seconds.

The link-state PDU refresh interval is derived from the link-state PDU lifetime and is equal to the lifetime minus 317 seconds.

Advertising Label-Switched Paths into IS-IS

You can advertise label-switched paths into IS-IS as point-to-point links, and the label-switched paths can be used in SPF calculations. The advertisement contains a local address (the from address of the label-switched path), a remote address (the to address of the label-switched path), and a metric with the following precedence:

- Use the label-switched path metric defined under IS-IS.
- Use the label-switched path metric configured for the label-switched path under MPLS.
- If you do not configure any of the above, use the default IS-IS metric of 10.

To advertise label-switched paths, include the label-switched-path statement, with a specified level and metric:

label-switched-path *name* level *level* metric *metric*;

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

NOTE: Before a single-hop label-switched path between a multiaccess link can be announced as up and used in SPF calculations, you must configure a label-switched path in both directions between two label-switched routers.

For more information about advertising label-switched paths, see the *JUNOS MPLS Applications Configuration Guide*.

Configuring IS-IS to Make Routing Devices Appear Overloaded

If the time elapsed after the IS-IS instance is enabled is less than the specified timeout, overload mode is set.

You configure or disable overload mode in IS-IS with or without a timeout. Without a timeout, overload mode is set until it is explicitly deleted from the configuration. With a timeout, overload mode is set if the time elapsed since the IS-IS instance started is less than the specified timeout.

A timer is started for the difference between the timeout and the time elapsed since the instance started. When the timer expires, overload mode is cleared. In overload

mode, the routing device IS-IS advertisements are originated with the overload bit set. This causes the transit traffic to avoid the overloaded routing device and take paths around the routing device. However, the overloaded routing device's own links are still accessible.

In overload mode, the routing device advertisement is originated with all the transit routing device links (except stub) set to a metric of 0xFFFF. The stub routing device links are advertised with the actual cost of the interfaces corresponding to the stub. This causes the transit traffic to avoid the overloaded routing device and take paths around the routing device. However, the overloaded routing device's own links are still accessible.

You can configure the local routing device so that it appears to be overloaded. You might want to do this when you want the routing device to participate in IS-IS routing, but do not want it to be used for transit traffic. (Note that traffic to immediately attached interfaces continues to transit the routing device.) To mark the routing device as overloaded, include the **overload** statement:

```
overload {
    advertise-high-metrics;
    timeout seconds;
}
```

To advertise maximum link metrics in network layer reachability information instead of setting the overload bit, include the **advertise-high-metrics** option when specifying the **overload** statement:

```
advertise-high-metrics;
```

To specify the number of seconds at which overload is reset, include the **timeout** option when specifying the **overload** statement:

```
overload timeout seconds;
```

The time can range from 60 through 1800 seconds.

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for these statements.

Configuring SPF Options for IS-IS

You can configure the following shortest-path-first (SPF) options:

- The delay in the time between the detection of a topology change and when the SPF algorithm actually runs.
- The maximum number of times that the SPF algorithm can run in succession before the hold-down timer begins.
- The time to hold down, or wait, before running another SPF calculation after the SPF algorithm has run in succession the configured maximum number of times.

To configure SPF options, include the **spf-options** statement:

```
spf-options {
    delay milliseconds;
    holddown milliseconds;
    rapid-runs number;
}
```

To configure the SPF delay, include the **delay** statement when specifying the **spf-options** statement:

```
delay milliseconds;
```

By default, the SPF algorithm runs 200 milliseconds after the detection of a topology change. The range that you can configure is from 50 through 1000 milliseconds.

To configure the maximum number of times that the SPF algorithm can run in succession, include the **rapid-runs** statement when specifying the **spf-options** statement:

```
rapid-runs number;
```

The default number of SPF calculations that can occur in succession is 3. The range that you can configure is from 1 through 5. Each SPF algorithm is run after the configured SPF delay. When the maximum number of SPF calculations occurs, the hold-down timer begins. Any subsequent SPF calculation is not run until the hold-down timer expires.

To configure the SPF hold-down timer, include the **holddown** statement when specifying the **spf-options** statement:

```
holddown milliseconds;
```

The default is 5000 milliseconds, and the range that you can configure is from 2000 through 10,000 milliseconds. Use the hold-down timer to hold down, or wait, before running any subsequent SPF calculations after the SPF algorithm runs for the configured maximum number of times. If the network stabilizes during the hold-down period and the SPF algorithm does not need to run again, the system reverts to the configured values for the **delay** and **rapid-runs** statements.

Configuring Graceful Restart for IS-IS

Graceful restart allows a routing device to restart with minimal effects to the network, and is enabled globally for all routing protocols at the **[edit routing-options]** hierarchy level. When graceful restart for IS-IS is enabled, the restarting routing device is not removed from the network topology during the restart period. The adjacencies are reestablished after restart is complete.

You can configure graceful restart parameters specifically for IS-IS. To do this, include the **graceful-restart** statement:

```
graceful-restart {
    helper-disable;
    restart-duration seconds;
}
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

To disable graceful restart for IS-IS, specify the disable statement. Helper mode is enabled by default. To disable the graceful restart helper capability, specify the helper-disable statement. To configure a time period for complete restart, specify the restart-duration statement. You can specify a number between 1 and 3600. The default value is 90 seconds.

Configuring IS-IS for Multipoint Network Clouds

IS-IS does not support multipoint configurations. Therefore, when configuring Frame Relay or Asynchronous Transfer Mode (ATM) networks, you must configure them as collections of point-to-point links, not as multipoint clouds.

Configuring IS-IS Traffic Engineering Attributes

You can configure the following IS-IS traffic engineering attributes:

- Configuring IS-IS to Use IGP Shortcuts on page 351
- Configuring IS-IS to Ignore the Metric of RSVP Label-Switched Paths on page 352
- Disabling IS-IS Support for Traffic Engineering on page 353
- Installing IPv4 Routes into the Multicast Routing Table on page 353
- Configuring IS-IS to Use Protocol Preference to Determine the Traffic Engineering Database Credibility Value on page 354

When configuring traffic engineering support, you can also configure IS-IS to use metric values greater than 63, as described in "Enabling Wide IS-IS Metrics for Traffic Engineering" on page 346.

Configuring IS-IS to Use IGP Shortcuts

IS-IS always performs SPF calculations to determine next hops. For prefixes reachable through a particular next hop, IS-IS places that next hop for that prefix in the inet.0 routing table. In addition, for routers running MPLS, IS-IS installs the prefix for IPv4 routes in the inet.3 routing table as well. The inet.3 table, which is present on the ingress router, contains the host address of each MPLS label-switched path (LSP) egress router. BGP uses this routing table to resolve next-hop addresses.

If you enable IS-IS traffic engineering shortcuts and if there is a label-switched path to a point along the path to that prefix, IS-IS installs the prefix in the inet.3 routing table and uses the LSP as a next hop. The net result is that for BGP egress routers for which there is no LSP, BGP automatically uses an LSP along the path to reach the egress router.

In JUNOS Release 9.3 and later, IS-IS traffic engineering shortcuts support IPv6 routes. LSPs to be used for shortcuts continue to be signaled using IPv4. However, by default, shortcut routes calculated through IPv6 routes are added to the inet6.3 routing table. The default behavior is for only BGP to use LSPs in its calculations. If you configure MPLS so that both BGP and interior gateway protocols use LSPs for forwarding traffic,

shortcut routes calculated through IPv6 are added to the inet6.0 routing table. IS-IS ensures that the IPv6 routes running over the IPv4 MPLS LSP are correctly de-encapsulated at the tunnel egress by pushing an extra IPv6 explicit null label between the IPv6 payload and the IPv4 transport label.

RSVP LSPs with a higher preference than IS-IS routes are not considered during the computation of traffic engineering shortcuts.

To configure IS-IS so that it uses label-switched paths as shortcuts when installing information in the inet.3 or inet6.3 routing table, include the following statements:

```
traffic-engineering {
    family inet {
        shortcuts;
        }
    }
    family inet6 {
        shortcuts;
    }
}
```

For IPv4 traffic, include the inet statement. For IPv6 traffic, include the inet6 statement.

To ignore the metric of RSVP LSPs in shortcut decisions, include the ignore-lsp-metrics statement:

```
traffic-engineering {
    ignore-lsp-metrics;
}
```

This option avoids mutual dependency between IS-IS and RSVP, eliminating the time period when the RSVP metric used for shortcuts is not up to date.

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for these statements.

Because the inet.3 routing table is present only on ingress routers, you can configure LSP shortcuts only on these routers.

For more information about configuring LSPs and MPLS, see the *JUNOS MPLS Applications Configuration Guide*.

Configuring IS-IS to Ignore the Metric of RSVP Label-Switched Paths

You can configure IS-IS to ignore the metric of RSVP label-switched paths (LSPs) when LDP tunneling is enabled. If you are using the RSVP for traffic engineering, you can run LDP simultaneously to eliminate the distribution of external routes in the core. The LSPs established by LDP are tunneled through the LSPs established by RSVP. LDP effectively treats the traffic-engineered LSPs as single hops. Ignoring the metric of RSVP LSPs avoids mutual dependency between IS-IS and RSVP, eliminating the time period when the RSVP metric used for tunneling traffic is not up to date.

To configure IS-IS to ignore the metric of RSVP LSPs, include the ignore-lsp-metrics statement:

```
traffic-engineering {
    ignore-lsp-metrics;
}
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

For more information about configuring LSPs and MPLS, see the *JUNOS MPLS Applications Configuration Guide*.

Disabling IS-IS Support for Traffic Engineering

By default, IS-IS supports traffic engineering by exchanging basic information with the traffic engineering database. To disable this support, and to disable IS-IS shortcuts if they are configured, include the disable statement:

```
traffic-engineering {
    disable;
}
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

Installing IPv4 Routes into the Multicast Routing Table

You can install unicast IPv4 routes into the multicast routing table (inet.2) for multicast reverse-path forwarding (RPF) checks.

To install routes into the multicast routing table for RPF checks, include the multicast-rpf-routes statement:

```
traffic-engineering {
    family inet {
        shortcuts {
            multicast-rpf-routes;
        }
    }
}
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

NOTE: Traffic engineering shortcuts must be enabled.

NOTE: IPv4 multicast topology must not be enabled.

NOTE: LSPs must not be advertised into IS-IS.

Configuring IS-IS to Use Protocol Preference to Determine the Traffic Engineering Database Credibility Value

By default, the JUNOS Software prefers IS-IS routes in the traffic engineering database over other IGP routes even if the routes of another IGP are configured with a lower, that is, more preferred, preference value. The traffic engineering database assigns a credibility value to each IGP and prefers the routes of the IGP with the highest credibility value. In JUNOS Release 9.4 and later, you can configure IS-IS to take protocol preference into account to determine the traffic engineering database credibility value. When protocol preference is used to determine the credibility value, IS-IS routes are not automatically preferred by the traffic engineering database, depending on your configuration. For example, OSPF routes have a default preference value of 10, while IS-IS Level 1 routes have a default preference value of 15. When protocol preference is enabled, the credibility value is determined by deducting the protocol preference value from a base value of 512. Using default protocol preference values, OSPF has a credibility value of 502, while IS-IS has a credibility value of 497. Because the traffic engineering database prefers IGP routes with the highest credibility value, OSPF routes are now preferred.

NOTE: This feature is also supported for OSPFv2. For more information, see "Enabling OSPF Traffic Engineering Support" on page 481.

To configure IS-IS to use the configured protocol preference for IGP routes to determine the traffic engineering database credibility value, include the credibility-protocol-preference statement at the [edit protocols isis traffic-engineering] hierarchy level:

```
[edit protocols isis]
traffic-engineering {
    credibility-protocol-preference;
}
```

Enabling Authentication for IS-IS Without Network-Wide Deployment

To allow the use of authentication without requiring network-wide deployment, include the loose-authentication-check statement:

```
loose-authentication-check;
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

Configuring Quicker Advertisement of IS-IS Adjacency State Changes

A hold-down timer delays the advertising of adjacencies by waiting until a time period has elapsed before labeling adjacencies in the up state. You can disable this hold-down timer, which labels adjacencies up faster. However, disabling the hold-down timer creates more frequent link-state PDU updates and SPF computation.

To disable the adjacency hold-down timer, include the no-adjacency-holddown statement:

```
no-adjacency-holddown;
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

Enabling Padding of IS-IS Hello Packets

You can configure padding on hello packets to accommodate asymmetrical maximum transfer units (MTUs) from different routing devices. This help prevents a premature adjacency UP state when one routing device's MTU does not meet the requirements to establish the adjacency.

As an OSI Layer 2 protocol, IS-IS does not support data fragmentation. Therefore, maximum packet sizes must be established and supported between two routers. During adjacency establishment, the IS-IS protocol makes sure that the link supports a packet size of 1,492 bytes by padding outgoing hello packets up to the maximum packet size of 1,492 bytes.

To configure padding for hello packets, include the hello-padding statement:

```
hello-padding (adaptive | loose | strict);
```

There are three types of hello padding:

- Adaptive padding. On point-to-point connections, the hello packets are padded from the initial detection of a new neighbor until the neighbor verifies the adjacency as Up in the adjacency state TLV. If the neighbor does not support the adjacency state TLV, then padding continues. On LAN connections, padding starts from the initial detection of a new neighbor until there is at least one active adjacency on the interface. Adaptive padding has more overhead than loose padding and is able to detect MTU asymmetry from one side of the connection. This one-sided detection may result in generation of extra LSPs that are flooded throughout the network. Specify the adaptive option to configure enough padding to establish an adjacency to neighbors.
- Loose padding (the default). The hello packet is padded from the initial detection of a new neighbor until the adjacency transitions to the Up state. Loose padding may not be able to detect certain situations such as asymmetrical MTUs between the routing devices. Specify the loose option to configure enough padding to initialize an adjacency to neighbors.
- Strict padding. Padding is done on all interface types and for all adjacency states, and is continuous. Strict padding has the most overhead. The advantage is that

strict padding detects MTU issues on both sides of a link. Specify the strict option to configure padding to allow all adjacency states with neighbors.

For a list of hierarchy levels at which you can include this statement, see the statement summary sections for this statement.

Configuring CLNS for IS-IS

Connectionless Network Services (CLNS) is a Layer 3 protocol, similar to IP version 4 (IPv4). CLNS uses network service access points (NSAPs) to address end systems and intermediate systems.

You can use IS-IS as the IGP to carry ISO CLNS routes through a network.

NOTE: CLNS is supported for the J Series Services Router only.

To enable IS-IS to exchange CLNS routes, include the clns-routing statement:

```
clns-routing;
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

You can configure a pure CLNS network by disabling IPv4 and IPv6 for IS-IS.

To disable IPv4, include the no-ipv4-routing statement:

```
no-ipv4-routing;
```

To disable IPv6, include the no-ipv6-routing statement:

```
no-ipv6-routing;
```

For a list of hierarchy levels at which you can include these statements, see the statement summary section for these statements.

You can export BGP routes into Layer 2 IS-IS by configuring an export policy and applying the policy to IS-IS. You can export BGP routes from a specific VRF instance into IS-IS by configuring and applying an export policy at the [edit routing-instance *instance-name* protocols isis] hierarchy level. ES-IS routes from one routing instance cannot be exported into a Layer 1 IS-IS area of another routing instance.

To configure an export policy to export BGP routes into IS-IS, include the policy-statement statement:

```
policy-statement policy-name {
    from {
        protocol bgp;
        family iso;
    }
    then {
```

```
        accept;
    }
}
```

To apply an export policy, include the **export** statement at the **[edit protocols isis]** hierarchy level:

```
export policy-name;
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for these statements.

For more information on policy configuration, see the *JUNOS Policy Framework Configuration Guide*.

You can also export routes from protocols other than BGP into IS-IS. ES-IS routes are exported to IS-IS by default. You can export ES-IS routes into IS-IS by configuring a routing policy.

For information on CLNS, see the *JUNOS Software Interfaces and Routing Configuration Guide*.

Example: Configuring CLNS for IS-IS

Configure a routing policy to accept CLNS routes:

```
policy-options {
    policy-statement dist-bgp {
        from {
            protocol bgp;
            family iso;
        }
        then accept;
    }
    policy-statement dist-static {
        from {
            protocol static;
            family iso;
        }
        then accept;
    }
}
```

Configure CLNS for IS-IS:

```
protocols {
    isis {
        traceoptions {
            file isis size 5m world-readable;
            flag error;
        }
        export dist-static;
        no-ipv6-routing;
        no-ipv4-routing;
        clns-routing;
```

```
        interface fe-0/0/1.0;
        interface t1-0/2/1.0;
        interface fxp0.0 {
            disable;
        }
        interface lo0.0;
    }
}
```

Configure a routing instance that supports CLNS routes:

```
routing-instances {
    aaaa {
        instance-type vrf;
        interface lo0.1;
        interface t1-3/0/0.0;
        interface fe-5/0/1.0;
        route-distinguisher 10.245.245.1:1;
        vrf-target target:11111:1;
        protocols {
            isis {
                export dist-bgp;
                no-ipv4-routing;
                no-ipv6-routing;
                clns-routing;
                interface all;
            }
        }
    }
}
```

Disabling IS-IS

To disable IS-IS on the routing device without removing the IS-IS configuration statements from the configuration, include the **disable** statement:

```
isis {
    disable;
}
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

To reenable IS-IS, remove the **disable** statement from the configuration:

```
[edit protocols]
user@host# delete isis disable
[edit protocols]
user@host# show
isis;
```

Disabling IPv4 Routing for IS-IS

You can disable IP version 4 (IPv4) routing for IS-IS. Disabling IPv4 routing results in the following:

- Routing device does not advertise the NLPID for IPv4 in JUNOS Software 0th link-state PDU fragment.
- Routing device does not advertise any IPv4 prefixes in JUNOS Software link-state PDUs.
- Routing device does not advertise the NLPID for IPv4 in JUNOS Software hello packets.
- Routing device does not advertise any IPv4 addresses in JUNOS Software hello packets.
- Routing device does not calculate any IPv4 routes.

To disable IPv4 routing on the routing device, include the no-ipv4-routing statement:

```
isis {
    no-ipv4-routing;
}
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

To reenable IS-IS, remove the no-ipv4-routing statement from the configuration:

```
[edit protocols]
user@host# delete isis no-ipv4-routing
```

Disabling IPv6 Routing for IS-IS

You can disable IP version 6 (IPv6) routing for IS-IS. Disabling IPv6 routing results in the following:

- Routing device does not advertise the NLPID for IPv6 in JUNOS Software 0th link-state PDU fragment.
- Routing device does not advertise any IPv6 prefixes in JUNOS Software link-state PDUs.
- Routing device does not advertise the NLPID for IPv6 in JUNOS Software hello packets.
- Routing device does not advertise any IPv6 addresses in JUNOS Software hello packets.
- Routing device does not calculate any IPv6 routes.

To disable IPv6 routing on the routing device, include the no-ipv6-routing statement:

```
isis {
```

```
    no-ipv6-routing;
}
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

To re-enable IS-IS, remove the disable statement from the configuration:

```
[edit protocols]
user@host# delete isis no-ipv6-routing
```

Applying Policies to Routes Exported to IS-IS

All routing protocols store the routes that they learn in the routing table. The routing table uses this collected route information to determine the active routes to destinations. The routing table then installs the active routes into its forwarding table and exports them into the routing protocols. It is these exported routes that the protocols advertise.

For each protocol, you control which routes the protocol stores in the routing table and which routes the routing table exports into the protocol from the routing table by defining a *routing policy* for that protocol. For information about defining routing policy, see the *JUNOS Policy Framework Configuration Guide*.

To apply routing policies that affect how the routing protocol process (rpd) exports routes into IS-IS, include the export statement:

```
export [ policy-names ];
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

NOTE: For IS-IS, you cannot apply routing policies that affect how routes are imported into the routing table; doing so with a link-state protocol can easily lead to an inconsistent topology database.

Examples: Configuring IS-IS Routing Policy

Define a policy that allows only host routes from USC (128.125.0.0/16), and apply the policy to routes exported from the routing table into IS-IS:

```
policy-options {
    policy-statement usc-hosts-only {
        term first {
            from {
                route-filter 128.125.0.0/16 upto /31;
            }
            then reject;
        }
        then accept;
    }
```

```
}
protocols {
    isis {
        export usc-hosts-only;
    }
}
```

Define a policy that takes BGP routes from the Edu community and places them into IS-IS with a metric of 14. Apply the policy to routes exported from the routing table into IS-IS:

```
protocols {
    isis {
        export edu-to-isis;
    }
}
policy-options {
    community Edu members 666:5;
    policy-statement edu-to-isis {
        from {
            protocol bgp;
            community Edu;
        }
        to protocol isis;
        then metric 14;
    }
}
```

Define a policy that rejects all IS-IS Level 1 routes so that none are exported into IS-IS:

```
policy-options {
    policy-statement level1 {
        term first {
            from level 1;
            then reject;
        }
        then accept;
    }
}
protocols {
    isis {
        export level1;
        interface fxp0;
    }
}
```

Define a routing policy to export IS-IS Level 1 internal-only routes into Level 2:

```
[edit]
protocols {
    isis {
        export L1-L2;
    }
}
policy-statement L1-L2 {
```

```
    term one {
        from {
            level 1;
            external;
        }
        then reject;
    }
    term two {
        from level 1;
        to level 2;
        then accept;
    }
}
```

Define a routing policy to export IS-IS Level 2 routes into Level 1:

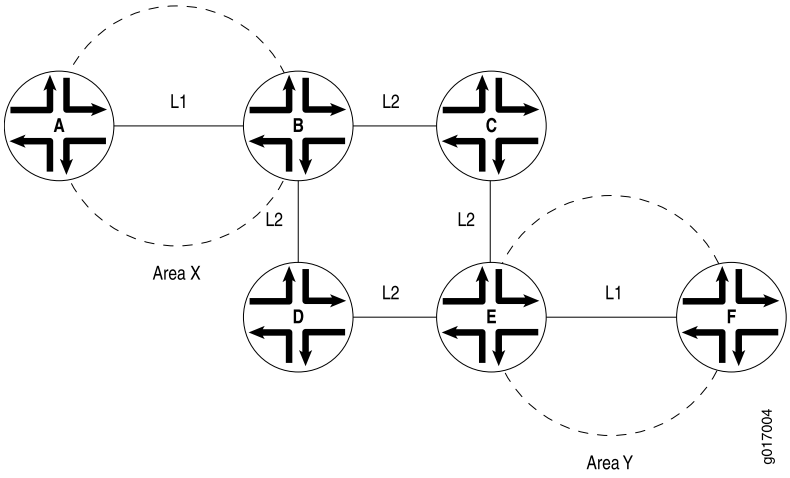
```
[edit]
protocols {
    isis {
        export L2-L1;
    }
}
policy-statement L2-L1 {
    term one {
        from level 2;
        to level 1;
        then accept;
    }
}
```

Installing a Default Route to the Nearest Routing Device That Operates at Both IS-IS Levels

When a routing device that operates as both a Level 1 and Level 2 router (Router B) determines that it can reach at least one area other than its own (for example, in Area Y), it sets the ATTACHED bit in its Level LSP. Thereafter, the Level 1 router (Router A) introduces a default route pointing to the nearest attached routing device that operates as both a Level 1 and Level 2 router (Router B). See Figure 7 on page 362.

Figure 7: Install Default Route to Nearest Routing Device That Operates at Both

Level 1 and Level 2



Configuring Loop-Free Alternate Routes for IS-IS

In JUNOS Release 9.5 and later, support for IS-IS loop-free alternate routes enables IP fast-reroute capability for IS-IS. The JUNOS Software precomputes loop-free backup routes for all IS-IS routes. These backup routes are preinstalled in the Packet Forwarding Engine, which performs a local repair and implements the backup path when the link for a primary next hop for a particular route is no longer available. With local repair, the Packet Forwarding Engine can correct a path failure before it receives recomputed paths from the Routing Engine. Local repair reduces the amount of time needed to reroute traffic to less than 50 milliseconds. In contrast, global repair can take up to 800 milliseconds to compute a new route. Local repair and global repair are thus complementary. Local repair enables traffic to continue to be routed using a backup path until global repair is able to calculate a new route.

A loop-free path is one that does not forward traffic back through the routing device to reach a given destination. That is, a neighbor whose shortest path to the destination traverses the routing device is not used as a backup route to that destination. To determine loop-free alternate paths for IS-IS routes, the JUNOS Software runs shortest-path-first (SPF) calculations on each one-hop neighbor. You can enable support for alternate loop-free routes on any IS-IS interface. Because it is common practice to enable LDP on an interface for which IS-IS is already enabled, this feature also provides support for LDP label-switched paths (LSPs).

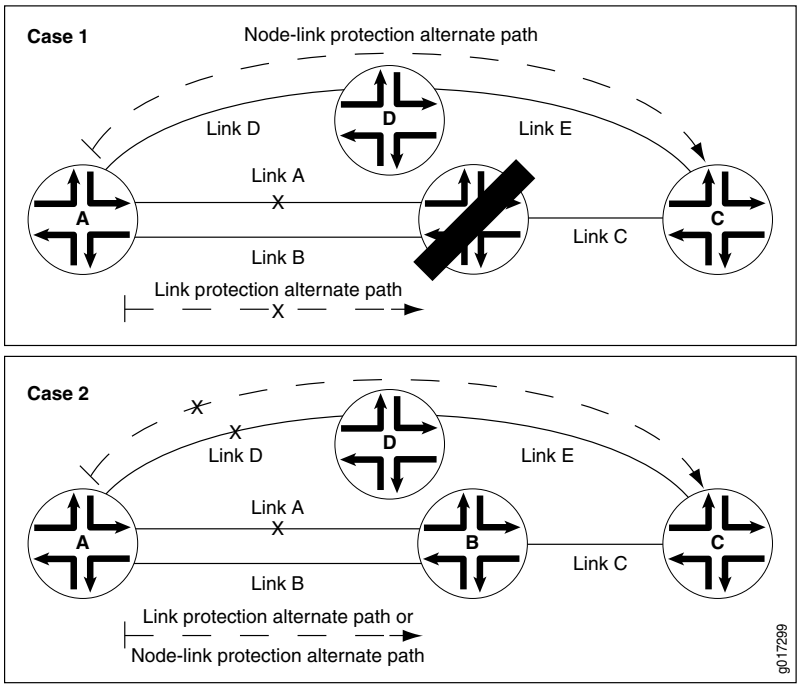
The level of backup coverage available through IS-IS routes depends on the actual network topology and is typically less than 100 percent for all destinations on any given routing device. You can extend backup coverage to include RSVP LSP paths.

The JUNOS Software provides two mechanisms for route redundancy for IS-IS through alternate loop-free routes: link protection and node-link protection. When you enable link protection or node-link protection on an IS-IS interface, the JUNOS Software creates an alternate path to the primary next hop for all destination routes that traverse a protected interface. Link protection offers per-link traffic protection. Use link protection when you assume that only a single link might become unavailable but that the neighboring node on the primary path would still be available through another interface.

Node-link protection establishes an alternate path through a different routing device altogether. Use node-link protection when you assume that access to a node is lost when a link is no longer available. As a result, the JUNOS Software calculates a backup path that avoids the primary next-hop routing device. In JUNOS Release 9.4 and earlier, only the RSVP protocol supports Packet Forwarding Engine local repair and fast reroute as well as link protection and node protection.

In Figure 8 on page 364, Case 1 shows how link protection allows source Router A to switch to Link B when the primary next hop Link A to destination Router C fails. However, if Router B fails, Link B also fails, and the protected Link A is lost. If node-link protection is enabled, Router A is able to switch to Link D on Router D and bypass the failed Router B altogether. As shown in Case 2, with node-link protection enabled, Link A on Router A has both link protection and node-link protection alternate paths available. That means that if the backup path from Router A to Link D fails, Link B remains available as an alternate backup path.

Figure 8: Link Protection and Node-Link Protection Comparison for IS-IS Routes



The JUNOS Software implementation of support for loop-free alternate paths for IS-IS routes is based on the following standards:

- Internet draft draft-ietf-rtgwg-ipfrr-spec-base-12.txt, *Basic Specification for IP Fast-Reroute: Loop-free Alternates*
- Internet draft draft-ietf-rtgwg-ipfrr-framework-06.txt, *IP Fast Reroute Framework*

This section discusses the following topics:

- Configuring Link Protection for IS-IS on page 365
- Configuring Node-Link Protection for IS-IS on page 365

- Excluding an IS-IS Interface as a Backup for Protected Interfaces on page 366
- Configuring RSVP Label-Switched Paths as Backup Paths for IS-IS on page 366
- Using Operational Mode Commands to Monitor Protected IS-IS Routes on page 367
- Example: Configuring Node-Link Protection for IS-IS Routes on page 367

Configuring Link Protection for IS-IS

You can configure link protection on any interface for which IS-IS is enabled. When you enable link protection, the JUNOS Software creates an alternate path to the primary next hop for all destination routes that traverse a protected interface. Link protection assumes that only a single link becomes unavailable but that the neighboring node would still be available through another interface.

NOTE: You must also configure a per-packet load-balancing routing policy to ensure that the routing protocol process installs all the next hops for a given route in the routing table. For more information, see "Example: Configuring Node-Link Protection for IS-IS Routes" on page 367 and "Configuring Per-Packet Load Balancing" on page 120.

To enable link protection, include the link-protection statement at the [edit protocols isis interface *interface-name*] hierarchy level:

```
[edit]
protocols {
   isis {
      interface interface-name:
         link-protection;
      }
   }
}
```

Configuring Node-Link Protection for IS-IS

You can configure node-link protection on any interface for which IS-IS is enabled. Node-link protection establishes an alternate path through a different routing device altogether for all destination routes that traverse a protected interface. Node-link protection assumes that the entire routing device, or node, has failed. The JUNOS Software therefore calculates a backup path that avoids the primary next-hop routing device.

NOTE: You must also configure a per-packet load-balancing routing policy to ensure that the routing protocol process installs all the next hops for a given route in the routing table. For more information, see "Example: Configuring Node-Link Protection for IS-IS Routes" on page 367 and "Configuring Per-Packet Load Balancing" on page 120.

To enable node-link protection, include the node-link-protection statement at the [edit protocols isis interface *interface-name*] hierarchy level:

```
[edit]
protocols {
   isis {
      interface interface-name:
         node-link-protection;
      }
   }
}
```

Excluding an IS-IS Interface as a Backup for Protected Interfaces

By default, all IS-IS interfaces that belong to the master instance or a specific routing instance are eligible as backup interfaces for protected interfaces. You can specify that any IS-IS interface be excluded from functioning as a backup interface to protected interfaces. To exclude an IS-IS interface as a backup interface, include the **no-eligible-backup** statement at the [**edit protocols isis interface** *interface-name*] hierarchy level:

```
[edit]
protocols {
   isis {
      interface interface-name {
         no-eligible-backup;
      }
   }
}
```

Configuring RSVP Label-Switched Paths as Backup Paths for IS-IS

Relying on the shortest-path first (SPF) calculation of backup paths for one-hop neighbors might result in less than 100 percent backup coverage for a specific network topology. You can enhance coverage of IS-IS and LDP label-switched paths (LSPs) by configuring RSVP LSPs as backup paths. To configure a specific RSVP LSP as a backup path, include the **backup** statement at the [**edit protocols mpls label-switched-path** *lsp-name*] hierarchy level:

```
[edit]
protocols {
   mpls {
      label-switched-path lsp-name {
         backup;
         to ip-address;
      }
   }
}
```

When configuring an LSP, you must specify the IP address of the egress routing device with the **to** statement. For detailed information about configuring LSPs and RSVP, see the *JUNOS MPLS Applications Configuration Guide*.

Using Operational Mode Commands to Monitor Protected IS-IS Routes

You can issue operational mode commands that provide more details about your link-protected and node-link-protected IS-IS routes. The following guidelines explain the type of information available from the output of each command:

- show isis backup label-switched-path—Displays which MPLS LSPs have been designated as backup paths and the current status of those LSPs.
- show isis backup spf results—Displays SPF calculations for each neighbor for a given destination. Indicates whether a specific interface or node has been designated as a backup path and why. Use the no-coverage option to display only those nodes that do not have backup coverage.
- show isis backup coverage—Displays the percentage of nodes and prefixes for each type of address family that are protected.
- show isis interface detail—Displays the type of protection (link or node-link) applied to each protected interface.

For more detailed information about these commands, see the *JUNOS Routing Protocols and Policies Command Reference*.

Example: Configuring Node-Link Protection for IS-IS Routes

In this example, all the logical interfaces on the router are enabled for IS-IS level 2, LDP, and RSVP. Node-link protection is enabled on all the interfaces, which means that if the primary next hop for any destination that traverses the interfaces becomes unavailable, the JUNOS Software uses a backup link that avoids the next-hop router altogether if necessary.

You also need to configure a routing policy that requires all traffic to use per-packet load balancing in order to enable Packet Forwarding Engine local repair. With local repair, the Packet Forwarding Engine can correct a path failure and implement a backup loop-free alternate route before it receives recomputed paths from the Routing Engine.

Configure the interfaces. Enable IS-IS and MPLS. In this example, the interfaces are also enabled for both IPv4 and IPv6 traffic.

```
[edit interfaces]
ge-2/0/0 {
   unit 0 {
      family inet {
         address 11.14.0.1/30;
      }
      family iso;
      family inet6;
      family mpls;
   }
}

ge-2/0/1 {
   unit 0 {
```

```
        family inet {
            address 11.14.1.1/30;
        }
        family iso;
        family inet6;
        family mpls;
    }
}

so-3/0/1 {
    unit 0 {
        family inet {
            address 11.16.1.1/30;
        }
        family iso;
        family inet6;
        family mpls;
    }
}

so-3/0/2 {
    unit 0 {
        family inet {
            address 11.16.0.1/30;
        }
        family iso;
        family inet6;
        family mpls;
    }
}

so-6/0/0 {
    unit 0 {
        family inet {
            address 11.12.0.1/30;
        }
        family iso;
        family inet6;
        family mpls;
    }
}
```

Configure the IS-IS interfaces for Level 2 only, and configure MPLS to use both RSVP and LDP label-switched paths (LSPs). Enable IS-IS node-link protection, which also automatically extends backup coverage to all LDP LSPs.

```
[edit protocols]
rsvp {
    interface all;
    interface fxp0.0 {
        disable;
    }
}
mpls {
    interface all;
    interface fxp0.0 {
```

```
        disable;
    }
}
isis {
    interface all {
        node-link-protection; # Enable node-link protection on all IS-IS interfaces.
            # Protection is automatically extended to all LDP LSPs.
        level 2 metric 10;
        level 1 disable;
    }
    interface fxp0.0 {
        disable;
    }
    interface lo0.0 {
        level 2 metric 0;
    }
}
ldp {
    deaggregate; # Enable forwarding equivalence class deaggregation, which results
        in faster global convergence.
    interface all;
    interface fxp0.0 {
        disable;
    }
}
```

To enable Packet Forwarding Engine local repair, establish a policy that forces the routing protocol process to install all the next hops for a given route. This policy ensures that the backup route is installed in the forwarding table used by the Packet Forwarding Engine to forward traffic to a given destination. After this policy is configured, export it to the neighboring routers with the **export** statement at the **[edit routing-options forwarding-table]** hierarchy level.

```
[edit policy-options]
policy-statement ecmp {
    term 1 {
        then {
            load-balance per-packet;
        }
    }
}

[edit routing-options]
forwarding-table {
    export ecmp;
}
```

Tracing IS-IS Protocol Traffic

You can trace various types of IS-IS protocol traffic to help debug IS-IS protocol issues. To trace IS-IS protocol traffic include the **traceoptions** statement at the **[edit protocols isis]** hierarchy level:

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

```
    flag flag <flag-modifier> <disable>;
}
```

You can specify the following IS-IS protocol-specific trace options using the flag statement:

- csn—Complete sequence number PDU (CSNP) packets
- error—Errored packets
- graceful-restart—Graceful restart operations
- hello—Hello packets
- ldp-synchronization—Synchronization between IS-IS and LDP
- lsp—Link-state PDU packets
- lsp-generation—Link-state PDU generation packets
- nsr-synchronization—NSR synchronization events
- packets—All IS-IS protocol packets
- psn—Partial sequence number PDU (PSNP) packets
- spf—Shortest-path-first (SPF) calculations

You can optionally specify one or more of the following flag modifiers:

- detail—Detailed trace information
- receive—Packets being received
- send—Packets being transmitted

NOTE: Use the flag modifier detail with caution as this may cause the CPU to become very busy.

Global tracing options are inherited from the configuration set by the traceoptions statement at the [edit routing-options] hierarchy level. You can override the following global trace options for the IS-IS protocol using the traceoptions flag statement included at the [edit protocols isis] hierarchy level:

- all—All tracing operations
- general—All normal operations and routing table changes (a combination of the normal and route trace operations)
- normal—Normal events
- policy—Policy processing
- route—Routing information
- state—State transitions
- task—Routing protocol task processing
- timer—Routing protocol timer processing

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

Examples: Tracing IS-IS Protocol Traffic

A common configuration traces SPF calculations, LSP calculations, normal protocol operations, and errors in protocol operation:

```
[edit]
protocols {
   isis {
      traceoptions {
         file isis-log size 1m files 10;
         flag spf;
         flag lsp;
         flag error;
         flag normal;
      }
   }
}
```

Trace only unusual or abnormal operations to the file routing-log, and trace detailed information about all IS-IS packets to the file isis-log:

```
[edit]
routing-options {
   traceoptions {
      file routing-log;
   }
}
protocols {
   isis {
      traceoptions {
         file isis-log size 10k files 5;
         flag csn detail;
         flag hello detail;
         flag lsp detail;
         flag psn detail;
      }
   }
}
```

Perform detailed tracing of mesh-group flooding:

```
[edit]
protocols {
   isis {
      traceoptions {
         file isis-log;
         flag lsp detail;
      }
   }
}
```

IS-IS LSP packets that contain errors are discarded by default. To log these errors, specify the error tracing operation:

```
[edit]
protocols {
    isis {
        traceoptions {
            file isis-log;
            flag error;
        }
    }
}
```

Related Topics

- traceoptions statement
- For more information about tracing and global tracing options, see Tracing Global Routing Protocol Operations on page 129.

Chapter 16

Summary of IS-IS Configuration Statements

The following sections explain each of the IS-IS configuration statements. The statements are organized alphabetically.

authentication-key

Syntax authentication-key *key*;

Hierarchy Level [edit logical-systems *logical-system-name* protocols isis level *level-number*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols isis level *level-number*],
[edit protocols isis level *level-number*],
[edit routing-instances *routing-instance-name* protocols isis level *level-number*]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Authentication key (password). Neighboring routing devices use the password to verify the authenticity of packets sent from this interface. For the key to work, you also must include the authentication-type statement.

All routing devices must use the same password. If you are using the JUNOS IS-IS software with another implementation of IS-IS, the other implementation must be configured to use the same password for the domain, the area, and all interfaces adjacent to the Juniper Networks routing device.

Default If you do not include this statement and the authentication-type statement, IS-IS authentication is disabled.

Options *key*—Authentication password. The password can be up to 1024 characters long.

Characters can include any ASCII strings. If you include spaces, enclose all characters in quotation marks (“ ”).

CAUTION: A simple password for authentication is truncated if it exceeds 254 characters.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring IS-IS Authentication on page 325

authentication-type

Syntax authentication-type *authentication*;

Hierarchy Level [edit logical-systems *logical-system-name* protocols isis level *level-number*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols isis level *level-number*],
[edit protocols isis level *level-number*],
[edit routing-instances *routing-instance-name* protocols isis level *level-number*]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Enable authentication and specify the authentication scheme for IS-IS. If you enable authentication, you must specify a password by including the authentication-key statement.

Default If you do not include this statement and the authentication-key statement, IS-IS authentication is disabled.

Options *authentication*—Authentication scheme:

- md5—Use HMAC authentication in combination with MD5. HMAC-MD5 authentication is defined in RFC 2104, *HMAC: Keyed-Hashing for Message Authentication*.
- simple—Use a simple password for authentication. The password is included in the transmitted packet, making this method of authentication relatively insecure. We recommend that you *not* use this authentication method.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics

- authentication-key
- no-authentication-check
- Configuring IS-IS Authentication on page 325

bfd-liveness-detection

Syntax

```
bfd-liveness-detection {
    authentication {
        algorithm algorithm-name;
        key-chain key-chain-name;
        loose-check;
    }
    detection-time {
        threshold milliseconds;
    }
    minimum-interval milliseconds;
    minimum-receive-interval milliseconds;
    no-adaptation;
    transmit-interval {
        threshold milliseconds;
        minimum-interval milliseconds;
    }
    multiplier number;
    version (1 | automatic);
}
```

Hierarchy Level

[edit logical-systems *logical-system-name* protocols isis interface *interface-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols isis interface *interface-name*],
[edit protocols isis interface *interface-name*],
[edit routing-instances *routing-instance-name* protocols isis interface *interface-name*]

Release Information

Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.
detection-time threshold and transmit-interval threshold options added in JUNOS Release 8.2.
Support for logical systems introduced in JUNOS Release 8.3.
no-adaptation statement introduced in JUNOS Release 9.0.
authentication algorithm, authentication key-chain, and authentication loose-check statements introduced in JUNOS Release 9.6.

Description

Configure bidirectional failure detection timers and authentication.

Options

authentication algorithm *algorithm-name* —Configure the algorithm used to authenticate the specified BFD session: simple-password, keyed-md5, keyed-sha-1, meticulous-keyed-md5, meticulous-keyed-sha-1.

authentication key-chain *key-chain-name*—Associate a security key with the specified BFD session using the name of the security keychain. The name you specify must match one of the keychains configured in the authentication-key-chains key-chain statement at the [edit security] hierarchy level.

authentication loose-check—(Optional) Configure loose authentication checking on the BFD session. Use only for transitional periods when authentication may not be configured at both ends of the BFD session.

detection-time threshold *milliseconds*—Configure a threshold. When the BFD session detection time adapts to a value equal to or greater than the threshold, a single trap and a single system log message are sent.

minimum-interval *milliseconds*—Configure the minimum intervals at which the local routing device transmits a hello packet and then expects to receive a reply from the neighbor with which it has established a BFD session.
Range: 1 through 255,000

minimum-receive-interval *milliseconds*—Configure only the minimum interval at which the local routing device expects to receive a reply from a neighbor with which it has established a BFD session.
Range: 1 through 255,000

multiplier *number*—Configure the number of hello packets not received by a neighbor that causes the originating interface to be declared down.
Range: 1 through 255
Default: 3

no-adaptation—Specify that BFD sessions not adapt to changing network conditions. We recommend that you not disable BFD adaptation unless it is preferable not to have BFD adaptation enabled in your network.

transmit-interval threshold *milliseconds*—Configure a threshold. When the BFD session transmit interval adapts to a value greater than the threshold, a single trap and a single system log message are sent. The interval threshold must be greater than the minimum transmit interval.
Range: 0 through 4,294,967,295 ($2^{32} - 1$)

transmit-interval minimum-interval *milliseconds*—Configure only the minimum interval at which the routing device sends hello packets to a neighbor with which it has established a BFD session.
Range: 1 through 255,000

version—Specify the BFD version to detect.
Range: 1 (BFD version 1), or automatic (autodetection)
Default: automatic

The remaining statements are explained separately.

Required Privilege Level

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

Related Topics

- Configuring BFD for IS-IS on page 328
- Configuring BFD Authentication for IS-IS on page 333

checksum

Syntax checksum;

Hierarchy Level [edit logical-systems *logical-system-name* protocols isis interface *interface-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols isis interface *interface-name*],
[edit protocols isis interface *interface-name*],
[edit routing-instances *routing-instance-name* protocols isis interface *interface-name*]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Enable checksum for packets on this interface. The checksum cannot be enabled with MD5 hello authentication on the same interface.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Enabling Packet Checksum on IS-IS Interfaces on page 336

clns-routing

Syntax clns-routing;

Hierarchy Level [edit protocols isis],
[edit routing-instances *routing-instance-name* protocols isis]

Release Information Statement introduced before JUNOS Release 7.4.

Description Enable IS-IS to exchange CLNS routes.

NOTE: CLNS is supported for the J Series Services Router only.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring CLNS for IS-IS on page 356
- *JUNOS Software Interfaces and Routing Configuration Guide*

csnp-interval

Syntax csnp-interval (*seconds* | disable);

Hierarchy Level [edit logical-systems *logical-system-name* protocols isis interface *interface-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols isis interface *interface-name*],
[edit protocols isis interface *interface-name*],
[edit routing-instances *routing-instance-name* protocols isis interface *interface-name*]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Configure the interval between complete sequence number (CSN) packets on a LAN interface.

Options disable—Do not send CSN packets on this interface.

seconds—Number of seconds between the sending of CSN packets.
Range: 1 through 65,535 seconds
Default: 10 seconds

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring the Transmission Frequency for CSNP Packets on IS-IS Interfaces on page 336

disable

See the following sections:

- disable (IS-IS) on page 381
- disable (LDP Synchronization) on page 382

disable (IS-IS)

Syntax disable;

Hierarchy Level
[edit logical-systems *logical-system-name* protocols isis],
[edit logical-systems *logical-system-name* protocols isis interface *interface-name*],
[edit logical-systems *logical-system-name* protocols isis interface *interface-name* level *level-number*],
[edit logical-systems *logical-system-name* protocols isis traffic-engineering],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols isis],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols isis interface *interface-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols isis interface *interface-name* level *level-number*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols isis traffic-engineering],
[edit protocols isis],
[edit protocols isis interface *interface-name*],
[edit protocols isis interface *interface-name* level *level-number*],
[edit protocols isis traffic-engineering],
[edit routing-instances *routing-instance-name* protocols isis],
[edit routing-instances *routing-instance-name* protocols isis interface *interface-name*],
[edit routing-instances *routing-instance-name* protocols isis interface *interface-name* level *level-number*],
[edit routing-instances *routing-instance-name* protocols isis traffic-engineering]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Disable IS-IS on the routing device, on an interface, or on a level. At the [edit protocols isis traffic-engineering] hierarchy level, disable IS-IS support for traffic engineering.

Enabling IS-IS on an interface (by including the interface statement at the [edit protocols isis] or the [edit routing-instances *routing-instance-name* protocols isis] hierarchy level), disabling it (by including the disable statement), and not actually having IS-IS run on an interface (by including the passive statement) are mutually exclusive states.

Default IS-IS is enabled for Level 1 and Level 2 routers on all interfaces on which an International Organization for Standardization (ISO) protocol family is enabled.

IS-IS support for traffic engineering is enabled.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Introduction to IS-IS on page 315
- Disabling IS-IS Support for Traffic Engineering on page 353
- Disabling IS-IS on page 358

disable (LDP Synchronization)

Syntax disable;

Hierarchy Level [edit logical-systems *logical-system-name* protocols isis interface *interface-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols isis interface *interface-name*],
[edit protocols isis interface *interface-name*],
[edit routing-instances *routing-instance-name* protocols isis interface *interface-name*]

Release Information Statement introduced in JUNOS Release 7.5.

Description Disable LDP for IS-IS.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring Synchronization Between LDP and IS-IS on page 336

export

Syntax export [*policy-names*];

Hierarchy Level [edit logical-systems *logical-system-name* protocols isis],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols isis],
[edit protocols isis],
[edit routing-instances *routing-instance-name* protocols isis]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Apply one or more policies to routes being exported from the routing table into IS-IS.

Options *policy-names*—Name of one or more policies.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Applying Policies to Routes Exported to IS-IS on page 360
- *JUNOS Policy Framework Configuration Guide*
- *JUNOS Software Interfaces and Routing Configuration Guide*

external-preference

Syntax external-preference *preference*;

Hierarchy Level [edit logical-systems *logical-system-name* protocols isis level *level-number*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols isis level *level-number*],
[edit protocols isis level *level-number*],
[edit routing-instances *routing-instance-name* protocols isis level *level-number*]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Configure the preference of external routes.

Options *preference*—Preference value.
Range: 0 through 4,294,967,295 ($2^{32} - 1$)
Default: 15 (for Level 1 internal routes), 18 (for Level 2 internal routes), 160 (for Level 1 external routes), 165 (for Level 2 external routes)

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- preference
- Configuring Preference Values for IS-IS Routes on page 347

family

Syntax

```
family inet {
    shortcuts {
        multicast-rpf-routes;
    }
}
family inet6 {
    shortcuts;
}
```

Hierarchy Level

[edit logical-systems *logical-system-name* protocols isis traffic-engineering],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols isis traffic-engineering]
[edit protocols isis traffic-engineering],
[edit routing-instances *routing-instance-name* protocols isis traffic-engineering],

Release Information

Statement introduced in JUNOS Release 9.3.

Description

Configure the address family for traffic engineering IS-IS interior gateway protocol (IGP) shortcuts. Support for IPv6 for IGP shortcuts introduced in JUNOS Release 9.3.

Options

inet—IPv4 address family

inet6—IPv6 address family

The remaining statements are explained separately.

Required Privilege Level

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

Related Topics

- Configuring IS-IS to Use IGP Shortcuts on page 351

graceful-restart

Syntax

```
graceful-restart {
    disable;
    helper-disable;
    restart-duration seconds;
}
```

Hierarchy Level [edit logical-systems *logical-system-name* protocols isis],
[edit protocols isis]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Configure graceful restart for IS-IS.

Options

disable—Disable graceful restart.

helper-disable—Disable graceful restart helper capability. Helper mode is enabled by default.

restart-duration *seconds*—Configure the time period for the restart to last, in seconds.
Range: 30 through 300 seconds
Default: 30 seconds

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics

- Configuring Graceful Restart on page 124
- Configuring Graceful Restart for IS-IS on page 350

hello-authentication-key

Syntax hello-authentication-key *password*;

Hierarchy Level [edit logical-systems *logical-system-name* protocols isis interface *interface-name* level *number*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols isis interface *interface-name* level *number*],
[edit protocols isis interface *interface-name* level *number*],
[edit routing-instances *routing-instance-name* protocols isis interface *interface-name* level *number*]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Configure an authentication key (password) for hello packets. Neighboring routing devices use the password to verify the authenticity of packets sent from an interface. For the key to work, you also must include the hello-authentication-type statement.

Default By default, hello authentication is not configured on an interface. However, if IS-IS authentication is configured, the hello packets are authenticated using the IS-IS authentication type and password.

Options *password*—Authentication password. The password can be up to 255 characters. Characters can include any ASCII strings. If you include spaces, enclose all characters in quotation marks (“ ”).

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- authentication-key
- authentication-type
- hello-authentication-type
- Configuring Authentication for IS-IS Hello Packets on page 343

hello-authentication-type

Syntax hello-authentication-type (md5 | simple);

Hierarchy Level [edit logical-systems *logical-system-name* protocols isis interface *interface-name* level *number*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols isis interface *interface-name* level *number*],
[edit protocols isis interface *interface-name* level *number*],
[edit routing-instances *routing-instance-name* protocols isis interface *interface-name* level *number*]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Enable authentication on an interface for hello packets. If you enable authentication on hello packets, you must specify a password by including the hello-authentication-key statement.

Default By default, hello authentication is not configured on an interface. However, if IS-IS authentication is configured, the hello packets are authenticated using the IS-IS authentication type and password.

Options md5—Specifies Message Digest 5 as the packet verification type.

simple—Specifies simple authentication as the packet verification type.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics

- authentication-key
- authentication-type
- hello-authentication-key
- Configuring Authentication for IS-IS Hello Packets on page 343

hello-interval

Syntax hello-interval *seconds*;

Hierarchy Level [edit logical-systems *logical-system-name* protocols isis interface *interface-name* level *level-number*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols isis interface *interface-name* level *level-number*],
[edit protocols isis interface *interface-name* level *level-number*],
[edit routing-instances *routing-instance-name* protocols isis interface *interface-name* level *level-number*]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Frequency with which the routing device sends hello packets out of an interface, in seconds.

Options *seconds*—Frequency of transmission for hello packets.
Range: 1 through 20,000 seconds
Default: 3 seconds (for designated intersystem [DIS] routers), 9 seconds (for non-DIS routers)

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring the Transmission Frequency for IS-IS Hello Packets on page 343
- hold-time

hello-padding

Syntax hello-padding (adaptive | loose | strict);

Hierarchy Level [edit logical-systems *logical-system-name* protocols isis interface *interface-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols isis interface *interface-name*],
[edit protocols isis interface *interface-name*],
[edit routing-instances *routing-instance-name* protocols isis interface *interface-name*]

Release Information Statement introduced in JUNOS Release 8.0.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Configure padding on hello packets to accommodate asymmetrical maximum transfer units (MTUs) from different hosts.

Options adaptive—Configure padding until state of neighbor adjacency is up.

loose—Configure padding until state of adjacency is initialized.

strict—Configure padding for all adjacency states.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Enabling Padding of IS-IS Hello Packets on page 355

hold-time

See the following sections:

- hold-time (IS-IS) on page 390
- hold-time (LDP Synchronization) on page 391

hold-time (IS-IS)

Syntax hold-time *seconds*;

Hierarchy Level [edit logical-systems *logical-system-name* protocols isis interface *interface-name* level *level-number*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols isis interface *interface-name* level *level-number*],
[edit protocols isis interface *interface-name* level *level-number*],
[edit routing-instances *routing-instance-name* protocols isis interface *interface-name* level *level-number*]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Set the length of time a neighbor considers this router to be operative (up) after receiving a hello packet. If the neighbor does not receiver another hello packet within the specified time, it marks this routing device as inoperative (down). The hold time itself is advertised in the hello packets.

Options *seconds*—Hold-time value, in seconds.

Range: 3 through 65,535 seconds, or 1 to send out hello packets every 333 milliseconds

Default: 9 seconds (for DIS routers), 27 seconds (for non-DIS routers; three times the default hello interval)

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics

- hello-interval
- Configuring the Delay Before IS-IS Neighbors Mark the Routing Device as Down on page 344

hold-time (LDP Synchronization)

Syntax hold-time *seconds*;

Hierarchy Level [edit logical-systems *logical-system-name* protocols isis interface *interface-name* ldp-synchronization],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols isis interface *interface-name* ldp-synchronization],
[edit protocols isis interface *interface-name* ldp-synchronization],
[edit routing-instances *routing-instance-name* protocols isis interface *interface-name* ldp-synchronization]

Release Information Statement introduced in JUNOS Release 7.5.

Description Configure the time period to advertise the maximum cost metric for a link that is not fully operational.

Options *seconds*—Hold-time value, in seconds.
Range: 1 through 65,535 seconds
Default: Infinity

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring Synchronization Between LDP and IS-IS on page 336

ignore-attached-bit

Syntax ignore-attached-bit;

Hierarchy Level [edit logical-systems *logical-system-name* protocols isis],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols isis],
[edit protocols isis],
[edit routing-instances *routing-instance-name* protocols isis]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Ignore the attached bit on IS-IS Level 1 routers. Configuring this statement allows the routing device to ignore the attached bit on incoming Level 1 LSPs. If the attached bit is ignored, no default route, which points to the routing device which has set the attached bit, is installed.

Default The ignore-attached-bit statement is disabled by default.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Installing a Default Route to the Nearest Routing Device That Operates at Both IS-IS Levels on page 362

ignore-lsp-metrics

Syntax ignore-lsp-metrics;

Hierarchy Level [edit logical-systems *logical-system-name* protocols isis traffic-engineering],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols isis traffic-engineering],
[edit protocols isis traffic-engineering],
[edit routing-instances *routing-instance-name* protocols isis traffic-engineering]

Release Information Statement introduced in JUNOS Release 8.0.

Description Ignore the metrics for RSVP label-switched paths in IS-IS traffic engineering shortcut calculations or when you configure LDP over RSVP label-switched paths.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- shortcuts
- Configuring IS-IS to Use IGP Shortcuts on page 351
- traffic-engineering
- *JUNOS MPLS Applications Configuration Guide*

interface

Syntax

```
interface (all | interface-name) {
    disable;
    bfd-liveness-detection {
        authentication {
            algorithm algorithm-name;
            key-chain key-chain-name;
            loose-check;
        }
        detection-time {
            threshold milliseconds;
        }
        minimum-interval milliseconds;
        minimum-receive-interval milliseconds;
        transmit-interval {
            threshold milliseconds;
            minimum-interval milliseconds;
        }
        multiplier number;
    }
    checksum;
    csnp-interval (seconds | disable);
    hello-padding (adaptive | loose | strict);
    ldp-synchronization {
        disable;
        hold-time seconds;
    }
    lsp-interval milliseconds;
    mesh-group (value | blocked);
    no-adjacency-holddown;
    no-ipv4-multicast;
    no-ipv6-multicast;
    no-ipv6-unicast;
    no-unicast-topology;
    passive;
    point-to-point;
    level level-number {
        disable;
        hello-authentication-type authentication;
        hello-authentication-key key;
        hello-interval seconds;
        hold-time seconds;
        ipv4-multicast-metric number;
        ipv6-multicast-metric number;
        ipv6-unicast-metric number;
        metric metric;
        passive;
        priority number;
        te-metric metric;
    }
}
```

Hierarchy Level

[edit logical-systems *logical-system-name* protocols isis],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols isis],
[edit protocols isis],
[edit routing-instances *routing-instance-name* protocols isis]

Release Information

Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description

Configure interface-specific IS-IS properties. To configure more than one interface, include the interface statement multiple times.

Enabling IS-IS on an interface (by including the interface statement at the [edit protocols isis] or the [edit routing-instances *routing-instance-name* protocols isis] hierarchy level), disabling it (by including the disable statement), and not actually having IS-IS run on an interface (by including the passive statement) are mutually exclusive states.

Options

all—Have the JUNOS Software create IS-IS interfaces automatically.

interface-name—Name of an interface. Specify the full interface name, including the physical and logical address components. For details about specifying interfaces, see the *JUNOS Network Interfaces Configuration Guide* and the *JUNOS Services Interfaces Configuration Guide*.

The remaining statements are explained separately.

Required Privilege Level

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

Related Topics
- Configuring of Interface-Specific IS-IS Properties on page 327

ipv4-multicast

Syntax ipv4-multicast;

Hierarchy Level [edit logical-systems *logical-system-name* protocols isis topologies],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols isis topologies],
[edit protocols isis topologies],
[edit routing-instances *routing-instance-name* protocols isis topologies]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Configure alternate IPv4 multicast topologies.

Default Multicast topologies are disabled.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring IS-IS Multicast Topologies on page 338

ipv4-multicast-metric

Syntax ipv4-multicast-metric *metric*;

Hierarchy Level [edit logical-systems *logical-system-name* protocols isis interface *interface-name* level *level-number*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols isis interface *interface-name* level *level-number*],
[edit protocols isis interface *interface-name* level *level-number*],
[edit routing-instances *routing-instance-name* protocols isis interface *interface-name* level *level-number*]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Specify the multicast topology metric value for the level.

Options *metric*—Metric value.
Range: 0 through 16,777,215

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring IS-IS Multicast Topologies on page 338

ipv6-multicast

Syntax ipv6-multicast;

Hierarchy Level [edit logical-systems *logical-system-name* protocols isis topologies],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols isis topologies],
[edit protocols isis topologies],
[edit routing-instances *routing-instance-name* protocols isis topologies]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Configure alternate IPv6 multicast topologies.

Default Multicast topologies are disabled.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring IS-IS Multicast Topologies on page 338

ipv6-multicast-metric

Syntax ipv6-multicast-metric *metric*;

Hierarchy Level [edit logical-systems *logical-system-name* protocols isis interface *interface-name* level *level-number*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols isis interface *interface-name* level *level-number*],
[edit protocols isis interface *interface-name* level *level-number*],
[edit routing-instances *routing-instance-name* protocols isis interface *interface-name* level *level-number*]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Specify the IPv6 alternate multicast topology metric value for the level.

Options *metric*—Metric value.
Range: 0 through 16,777,215

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring IS-IS Multicast Topologies on page 338

ipv6-unicast

Syntax ipv6-unicast;

Hierarchy Level [edit logical-systems *logical-system-name* protocols isis topologies],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols isis topologies],
[edit protocols isis topologies],
[edit routing-instances *routing-instance-name* protocols isis topologies]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Configure alternate IPv6 unicast topologies.

Default IPv6 unicast topologies are disabled.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring IS-IS IPv6 Unicast Topologies on page 340

ipv6-unicast-metric

Syntax ipv6-unicast-metric *metric*;

Hierarchy Level [edit logical-systems *logical-system-name* protocols isis interface *interface-name* level *level-number*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols isis interface *interface-name* level *level-number*],
[edit protocols isis interface *interface-name* level *level-number*],
[edit routing-instances *routing-instance-name* protocols isis interface *interface-name* level *level-number*]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Specify the IPv6 unicast topology metric value for the level.

Options *metric*—Metric value.
Range: 0 through 16,777,215

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring IS-IS IPv6 Unicast Topologies on page 340

isis

Syntax isis { ... }

Hierarchy Level [edit logical-systems *logical-system-name* protocols],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols],
[edit protocols],
[edit routing-instances *routing-instance-name* protocols]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Enable IS-IS routing on the routing device or for a routing instance.

The isis statement is the one statement you must include in the configuration to run IS-IS on the routing device or in a routing instance.

Default IS-IS is disabled on the routing device.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Minimum IS-IS Configuration on page 324

label-switched-path

Syntax label-switched-path *name* level *level-number* metric *metric*;

Hierarchy Level [edit logical-systems *logical-system-name* protocols isis],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols isis],
[edit protocols isis],
[edit routing-instances *routing-instance-name* protocols isis]

Release Information Statement introduced before JUNOS Release 7.4.

Description Advertise LSPs into IS-IS as point-to-point links. The LSP is advertised in the appropriate IS-IS levels as a point-to-point link and contains a local address and a remote address.

Options *name*—Identifies the LSP.

level-number—IS-IS level number.
Values: 1 or 2

metric—Metric value.
Range: 1 through 63, or 1 through 16,777,215 (if you have configured wide metrics)
Default: 0 (for lo0), 10 (for all other interfaces)

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Advertising Label-Switched Paths into IS-IS on page 348

ldp-synchronization

Syntax

```
ldp-synchronization {
    disable;
    hold-time seconds;
}
```

Hierarchy Level

[edit logical-systems *logical-system-name* protocols isis interface *interface-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols isis interface *interface-name*],
[edit protocols isis interface *interface-name*],
[edit routing-instances *routing-instance-name* protocols isis interface *interface-name*]

Release Information

Statement introduced in JUNOS Release 7.5.

Description

Enable synchronization by advertising the maximum cost metric until LDP is operational on the link.

The remaining statements are explained separately.

Required Privilege Level

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

Related Topics

- Configuring Synchronization Between LDP and IGPs on page 476

level

See the following sections:

- level (Global IS-IS) on page 402
- level (IS-IS Interfaces) on page 403

level (Global IS-IS)

Syntax

```
level level-number {
    authentication-key key;
    authentication-type type;
    external-preference preference;
    no-csnp-authentication;
    no-hello-authentication;
    no-psnp-authentication;
    preference preference;
    wide-metrics-only;
}
```

Hierarchy Level

[edit logical-systems *logical-system-name* protocols isis],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols isis],
[edit protocols isis],
[edit routing-instances *routing-instance-name* protocols isis]

Release Information

Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description

Configure the global-level properties.

Options

level-number—IS-IS level number.

Values: 1 or 2

The remaining statements are explained separately.

Required Privilege Level

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

Related Topics

- Configuring Preference Values for IS-IS Routes on page 347

level (IS-IS Interfaces)

Syntax

```
level level-number {
    disable;
    hello-authentication-key key;
    hello-authentication-type authentication;
    hello-interval seconds;
    hold-time seconds;
    ipv4-multicast-metric number;
    ipv6-unicast-metric number;
    metric metric;
    passive;
    priority number;
    te-metric metric;
}
```

Hierarchy Level

[edit logical-systems *logical-system-name* protocols isis interface *interface-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols isis interface *interface-name*],
[edit protocols isis interface *interface-name*],
[edit routing-instances *routing-instance-name* protocols isis interface *interface-name*]

Release Information

Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description

Configure the IS-IS level. You can configure one instance of Level 1 routing and one instance of Level 2 routing on each interface, and you can configure the two levels differently.

Options

level-number—IS-IS level number.

Values: 1 or 2

Default: The routing device operates as both a Level 1 and Level 2 router.

The remaining statements are explained separately.

Required Privilege Level

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

Related Topics

- Configuring Levels on IS-IS Interfaces on page 341

link-protection

Syntax link-protection;

Hierarchy Level [edit logical-systems *logical-system-name* protocols isis interface *interface-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols isis interface *interface-name*],
[edit protocols isis interface *interface-name*],
[edit routing-instances *routing-instance-name* protocols isis interface *interface-name*]

Release Information Statement introduced in JUNOS Release 9.5.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Enable link protection on the specified IS-IS interface. The JUNOS Software creates a backup loop-free alternate path to the primary next hop for all destination routes that traverse the protected interface.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- node-link-protection
- Configuring Link Protection for IS-IS on page 365

loose-authentication-check

Syntax loose-authentication-check;

Hierarchy Level [edit logical-systems *logical-system-name* protocols isis],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols isis],
[edit protocols isis],
[edit routing-instances *routing-instance-name* protocols isis]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Allow the use of MD5 authentication without requiring network-wide deployment.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Enabling Authentication for IS-IS Without Network-Wide Deployment on page 354

lsp-interval

Syntax lsp-interval *milliseconds*;

Hierarchy Level [edit logical-systems *logical-system-name* protocols isis interface *interface-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols isis interface *interface-name*],
[edit protocols isis interface *interface-name*],
[edit routing-instances *routing-instance-name* protocols isis interface *interface-name*]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Configure the link-state PDU interval time.

Options *milliseconds*—Number of milliseconds between the sending of link-state PDUs. Specifying a value of 0 blocks all link-state PDU transmission.
Range: 0 through 1000 milliseconds
Default: 100 milliseconds

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring the Transmission Frequency for Link-State PDUs on IS-IS Interfaces on page 337

lsp-lifetime

Syntax lsp-lifetime *seconds*;

Hierarchy Level [edit logical-systems *logical-system-name* protocols isis],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols isis],
[edit protocols isis],
[edit routing-instances *routing-instance-name* protocols isis]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Specify how long a link-state PDU originating from the routing device should persist in the network. The routing device sends link-state PDUs often enough so that the link-state PDU lifetime never expires.

Options *seconds*—link-state PDU lifetime, in seconds.
Range: 350 through 65,535 seconds
Default: 1200 seconds

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring Link-State PDU Lifetime for IS-IS on page 347

max-areas

Syntax max-areas *number*;

Hierarchy Level [edit logical-systems *logical-system-name* protocols isis],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols isis]
[edit protocols isis],
[edit routing-instances *routing-instance-name* protocols isis]

Release Information Statement introduced in JUNOS Release 8.1.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Modify the maximum number of IS-IS areas advertised.

Options *number*—Maximum number of areas to include in the IS-IS hello (IIH) PDUs and link-state PDUs.
Range: 3 through 36
Default: 3

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Limiting the Number of Advertised IS-IS Areas on page 346

mesh-group

Syntax mesh-group (blocked | *value*);

Hierarchy Level [edit logical-systems *logical-system-name* protocols isis interface *interface-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols isis interface *interface-name*],
[edit protocols isis interface *interface-name*],
[edit routing-instances *routing-instance-name* protocols isis interface *interface-name*]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Configure an interface to be part of a mesh group, which is a set of fully connected nodes.

Options blocked—Configure the interface so that it does not flood link-state PDU packets.

value—Number that identifies the mesh group.
Range: 1 through 4,294,967,295 (2^{32} – 1; 32 bits are allocated to identify a mesh group)

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring Mesh Groups of IS-IS Interfaces on page 337

metric

Syntax metric *metric*;

Hierarchy Level [edit logical-systems *logical-system-name* protocols isis interface *interface-name* level *level-number*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols isis interface *interface-name* level *level-number*],
[edit protocols isis interface *interface-name* level *level-number*],
[edit routing-instances *routing-instance-name* protocols isis interface *interface-name* level *level-number*]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Specify the metric value for the level.

Options *metric*—Metric value.

Range: 1 through 63, or 1 through 16,777,215 (if you have configured wide metrics)

Default: 10 (for all interfaces except lo0), 0 (for the lo0 interface)

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics

- te-metric
- wide-metrics-only
- Configuring the Metric Value for IS-IS Routes on page 344

multicast-rpf-routes

Syntax multicast-rpf-routes;

Hierarchy Level [edit logical-systems *logical-system-name* protocols isis traffic-engineering family inet shortcuts],
[edit logical-systems *logical-system-name* routing-instances traffic-engineering family inet shortcuts],
[edit protocols isis traffic-engineering family inet shortcuts],
[edit routing-instances *routing-instance-name* protocols isis traffic-engineering family inet shortcuts]

Release Information Statement introduced in JUNOS Release 9.3.

Description Install IPv4 routes into the multicast routing table for RPF checks.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Installing IPv4 Routes into the Multicast Routing Table on page 353

no-adjacency-holddown

Syntax no-adjacency-holddown;

Hierarchy Level [edit logical-systems *logical-system-name* protocols isis],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols isis],
[edit protocols isis],
[edit routing-instances *routing-instance-name* protocols isis]

Release Information Statement introduced in JUNOS Release 8.0.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Disable the hold-down timer for IS-IS adjacencies.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring Quicker Advertisement of IS-IS Adjacency State Changes on page 355

no-authentication-check

Syntax no-authentication-check;

Hierarchy Level [edit logical-systems *logical-system-name* protocols isis],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols isis],
[edit protocols isis],
[edit routing-instances *routing-instance-name* protocols isis]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Generate authenticated packets and check the authentication on received packets, but do not reject packets that cannot be authenticated.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- csnp-interval
- hello-authentication-type
- Configuring IS-IS Authentication on page 325

no-csnp-authentication

Syntax no-csnp-authentication;

Hierarchy Level [edit logical-systems *logical-system-name* protocols isis level *level-number*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols isis level *level-number*],
[edit protocols isis level *level-number*],
[edit routing-instances *routing-instance-name* protocols isis level *level-number*]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Suppress authentication check on complete sequence number PDU (CSNP) packets.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- csnp-interval
- Configuring IS-IS Authentication on page 325

no-eligible-backup

Syntax no-eligible-backup;

Hierarchy Level [edit logical-systems *logical-system-name* protocols isis interface *interface-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols isis interface *interface-name*],
[edit protocols isis interface *interface-name*],
[edit routing-instances *routing-instance-name* protocols isis interface *interface-name*]

Release Information Statement introduced in JUNOS Release 9.5.
Statement introduced in JUNOS Release 9.5 for EX Series switches.

Description Exclude the specified interface as a backup interface for IS-IS interfaces on which link protection or node-link protection is enabled.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- link-protection
- node-link-protection
- Excluding an IS-IS Interface as a Backup for Protected Interfaces on page 366

no-hello-authentication

Syntax no-hello-authentication;

Hierarchy Level [edit logical-systems *logical-system-name* protocols isis level *level-number*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols isis level *level-number*],
[edit protocols isis level *level-number*],
[edit routing-instances *routing-instance-name* protocols isis level *level-number*]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Suppress authentication check on complete sequence number hello packets.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- hello-authentication-type
- Configuring IS-IS Authentication on page 325

no-ipv4-multicast

Syntax no-ipv4-multicast;

Hierarchy Level [edit logical-systems *logical-system-name* protocols isis interface *interface-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols isis interface *interface-name*],
[edit protocols isis interface *interface-name*],
[edit routing-instances *routing-instance-name* protocols isis interface *interface-name*]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Exclude an interface from the IPv4 multicast topologies.

Default Multicast topologies are disabled.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring IS-IS Multicast Topologies on page 338

no-ipv4-routing

Syntax no-ipv4-routing;

Hierarchy Level [edit logical-systems *logical-system-name* protocols isis],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols isis],
[edit protocols isis],
[edit routing-instances *routing-instance-name* protocols isis]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Disable IP version 4 (IPv4) routing.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Disabling IPv4 Routing for IS-IS on page 359

no-ipv6-multicast

Syntax no-ipv6-multicast;

Hierarchy Level [edit logical-systems *logical-system-name* protocols isis interface *interface-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols isis interface *interface-name*],
[edit protocols isis interface *interface-name*],
[edit routing-instances *routing-instance-name* protocols isis interface *interface-name*]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Exclude an interface from the IPv6 multicast topologies.

Default Multicast topologies are disabled.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring IS-IS Multicast Topologies on page 338

no-ipv6-routing

Syntax no-ipv6-routing;

Hierarchy Level [edit logical-systems *logical-system-name* protocols isis],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols isis],
[edit protocols isis],
[edit routing-instances *routing-instance-name* protocols isis]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Disable IP version 6 (IPv6) routing.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Disabling IPv6 Routing for IS-IS on page 359

no-ipv6-unicast

Syntax no-ipv6-unicast;

Hierarchy Level [edit logical-systems *logical-system-name* protocols isis interface *interface-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols isis interface *interface-name*],
[edit protocols isis interface *interface-name*],
[edit routing-instances *routing-instance-name* protocols isis interface *interface-name*]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Exclude an interface from the IPv6 unicast topologies.

Default IPv6 unicast topologies are disabled.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring IS-IS IPv6 Unicast Topologies on page 340

no-psnp-authentication

Syntax no-psnp-authentication;

Hierarchy Level [edit logical-systems *logical-system-name* protocols isis level *level-number*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols isis level *level-number*],
[edit protocols isis level *level-number*],
[edit routing-instances *routing-instance-name* protocols isis level *level-number*]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Suppress authentication check on partial sequence number PDU (PSNP) packets.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring IS-IS Authentication on page 325

no-unicast-topology

Syntax no-unicast-topology;

Hierarchy Level [edit logical-systems *logical-system-name* protocols isis interface *interface-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols isis interface *interface-name*],
[edit protocols isis interface *interface-name*],
[edit routing-instances *routing-instance-name* protocols isis interface *interface-name*]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Exclude an interface from the IPv4 unicast topologies.

Default IPv4 unicast topologies are disabled.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring IS-IS Multicast Topologies on page 338

node-link-protection

Syntax node-ink-protection;

Hierarchy Level [edit logical-systems *logical-system-name* protocols isis interface *interface-name*],
[edit logical-routers *logical-router-name* routing-instances *routing-instance-name* protocols isis interface *interface-name*],
[edit protocols isis interface *interface-name*],
[edit routing-instances *routing-instance-name* protocols isis interface *interface-name*]

Release Information Statement introduced in JUNOS Release 9.5.
Statement introduced in JUNOS Release 9.5 for EX Series switches.

Description Enable node-link protection on the specified IS-IS interface. The JUNOS Software creates an alternate loop-free path to the primary next hop for all destination routes that traverse a protected interface. This alternate path avoids the primary next-hop routing device altogether and establishes a path through a different routing device.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- link-protection
- Configuring Node-Link Protection for IS-IS on page 365

overload

Syntax

```
overload {
    advertise-high-metrics;
    timeout seconds;
}
```

Hierarchy Level

[edit logical-systems *logical-system-name* protocols isis],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols isis],
[edit protocols isis],
[edit routing-instances *routing-instance-name* protocols isis]

Release Information

Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description

Configure the local routing device so that it appears to be overloaded. You might want to do this when you want the routing device to participate in IS-IS routing, but do not want it to be used for transit traffic. Note that traffic to immediately attached interfaces continues to transit the routing device. You can also advertise maximum link metrics in network layer reachability information (NLRI) instead of setting the overload bit.

NOTE: If the time elapsed after the IS-IS instance is enabled is less than the specified timeout, overload mode is set.

Options

advertise-high-metrics—Advertise maximum link metrics in NLRIs instead of setting the overload bit.

timeout *seconds*—Number of seconds at which the overloading is reset.
Default: 0 seconds
Range: 60 through 1800 seconds

Required Privilege Level

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

Related Topics

- Configuring IS-IS to Make Routing Devices Appear Overloaded on page 348

passive

Syntax passive;

Hierarchy Level [edit logical-systems *logical-system-name* protocols isis interface *interface-name*],
[edit logical-systems *logical-system-name* protocols isis interface *interface-name* level *level-number*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols isis interface *interface-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols isis interface *interface-name* level *level-number*],
[edit protocols isis interface *interface-name*],
[edit protocols isis interface *interface-name* level *level-number*],
[edit routing-instances *routing-instance-name* protocols isis interface *interface-name*],
[edit routing-instances *routing-instance-name* protocols isis interface *interface-name* level *level-number*]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Advertise the direct interface addresses on an interface or into a level on the interface without actually running IS-IS on that interface or level.

This statement effectively prevents IS-IS from running on the interface. To enable IS-IS on an interface, include the interface statement at the [edit protocols isis] or the [edit routing-instances *routing-instance-name* protocols isis] hierarchy level. To disable it, include the disable statement at those hierarchy levels. The three states are mutually exclusive.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- disable
- Advertising Interface Addresses Without Running IS-IS on page 343

point-to-point

Syntax point-to-point;

Hierarchy Level [edit logical-systems *logical-system-name* protocols isis interface *interface-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols isis interface *interface-name*],
[edit protocols isis interface *interface-name*],
[edit routing-instances *routing-instance-name* protocols isis interface *interface-name*]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Configure an IS-IS interface to behave like a point-to-point connection.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring Point-to-Point Interfaces for IS-IS on page 340

preference

Syntax preference *preference*;

Hierarchy Level [edit logical-systems *logical-system-name* protocols isis level *level-number*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols isis level *level-number*],
[edit protocols isis level *level-number*],
[edit routing-instances *routing-instance-name* protocols isis level *level-number*]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Configure the preference of internal routes.

Options *preference*—Preference value.
Range: 0 through 4,294,967,295 ($2^{32} - 1$)
Default: 15 (for Level 1 internal routes), 18 (for Level 2 internal routes), 160 (for Level 1 external routes), 165 (for Level 2 external routes)

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- external-preference
- Configuring Preference Values for IS-IS Routes on page 347

prefix-export-limit

Syntax prefix-export-limit *number*;

Hierarchy Level [edit logical-systems *logical-system-name* protocols isis level *level-number*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols isis level *level-number*],
[edit protocols isis level *level-number*],
[edit routing-instances *routing-instance-name* protocols isis level *level-number*]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Configure a limit to the number of prefixes exported into IS-IS.

Options *number*—Prefix limit.
Range: 0 through 4,294,967,295 ($2^{32} - 1$)

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Limiting the Number of Prefixes Exported to IS-IS on page 347

priority

Syntax priority *number*;

Hierarchy Level [edit logical-systems *logical-system-name* protocols isis interface *interface-name* level *level-number*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols isis interface *interface-name* level *level-number*],
[edit protocols isis interface *interface-name* level *level-number*],
[edit routing-instances *routing-instance-name* protocols isis interface *interface-name* level *level-number*]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description The interface's priority for becoming the designated router. The interface with the highest priority value becomes that level's designated router.

The priority value is meaningful only on a multiaccess network. It has no meaning on a point-to-point interface.

Options *number*—Priority value.
Range: 0 through 127
Default: 64

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring the Designated Router Priority for IS-IS on page 345

reference-bandwidth

Syntax reference-bandwidth *reference-bandwidth*;

Hierarchy Level [edit logical-systems *logical-system-name* protocols isis],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols isis],
[edit protocols isis],
[edit routing-instances *routing-instance-name* protocols isis]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Set the reference bandwidth used in calculating the default interface cost. The cost is calculated using the following formula:

```
cost = reference-bandwidth/bandwidth
```

Options *reference-bandwidth*—Reference bandwidth, in megabits per second.
Default: 10 Mbps
Range: 9600 through 1,000,000,000,000 Mbps

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring the Reference Bandwidth Used in IS-IS Metric Calculations on page 345

rib-group

Syntax

```
rib-group {
    inet group-name;
    inet6 group-name;
}
```

Hierarchy Level

[edit logical-systems *logical-system-name* protocols isis],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols isis],
[edit protocols isis],
[edit routing-instances *routing-instance-name* protocols isis]

Release Information

Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description

Install routes learned from IS-IS routing instances into routing tables in the IS-IS routing table group. You can install IPv4 routes or IPv6 routes.

Support for IPv6 routing table groups in IS-IS enables IPv6 routes that are learned from IS-IS routing instances to be installed into other routing tables defined in an IS-IS routing table group.

Options

group-name—Name of the routing table group.

inet—Install IPv4 IS-IS routes.

inet6—Install IPv6 IS-IS routes.

Required Privilege Level

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

Related Topics

- Creating Routing Table Groups on page 114
- Configuring How Interface Routes Are Imported into Routing Tables on page 116
- Configuring BGP Routing Table Groups on page 772

shortcuts

Syntax

```
shortcuts {
    multicast-rpf-routes;
}
```

Hierarchy Level

[edit logical-systems *logical-system-name* protocols isis traffic-engineering family (inet | inet6)],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols isis traffic-engineering family (inet | inet6)],
[edit protocols isis traffic-engineering family (inet | inet6)],
[edit routing-instances *routing-instance-name* protocols isis traffic-engineering family (inet | inet6)]

Release Information

Statement introduced before JUNOS Release 7.4.
The family statement and support for IPv6 routes for IS-IS traffic engineering shortcuts introduced in JUNOS Release 9.3.

Description

Configure IS-IS to use MPLS label-switched paths (LSPs) as next hops if possible when installing routing information into the inet.3 or inet6.3 routing table.

The remaining statement is explained separately.

Required Privilege Level

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

Related Topics

- Configuring IS-IS to Use IGP Shortcuts on page 351

spf-options

Syntax

```
spf-options {
    delay milliseconds;
    holddown milliseconds;
    rapid-runs number;
}
```

Hierarchy Level

[edit logical-systems *logical-system-name* protocols isis],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols isis],
[edit protocols isis],
[edit routing-instances *routing-instance-name* protocols isis]

Release Information

Statement introduced in JUNOS Release 8.5.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description

Configure options for running the shortest-path-first (SPF) algorithm. You can configure a delay for when to run the SPF algorithm after a network topology change is detected, the maximum number of times the SPF algorithm can run in succession, and a holddown interval after SPF algorithm runs the maximum number of times.

Options

delay *milliseconds*—Time interval between the detection of a topology change and when the SPF algorithm runs.
Range: 50 through 1000 milliseconds
Default: 200 milliseconds

holddown *milliseconds*—Time interval to hold down, or wait before a subsequent SPF algorithm runs after the SPF algorithm has run the configured maximum number of times in succession.
Range: 2000 through 10,000 milliseconds
Default: 5000 milliseconds

rapid-runs *number*—Maximum number of times the SPF algorithm can run in succession. After the maximum is reached, the holddown interval begins.
Range: 1 through 5
Default: 3

Required Privilege Level

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

Related Topics

- Configuring SPF Options for IS-IS on page 349

te-metric

Syntax te-metric *metric*;

Hierarchy Level [edit logical-systems *logical-system-name* protocols isis interface *interface-name* level *level-number*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols isis interface *interface-name* level *level-number*],
[edit protocols isis interface *interface-name* level *level-number*],
[edit routing-instances *routing-instance-name* protocols isis interface *interface-name* level *level-number*]

Release Information Statement introduced before JUNOS Release 7.4.

Description Metric value used by traffic engineering for information injected into the traffic engineering database. The value of the traffic engineering metric does not affect normal IS-IS forwarding.

Options *metric*—Metric value.
Range: 1 through 16,777,215
Default: Value of the IGP metric

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- metric
- wide-metrics-only
- Configuring the Metric Value for IS-IS Routes on page 344

topologies

Syntax

```
topologies {
    ipv4-multicast;
    ipv6-multicast;
    ipv6-unicast;
}
```

Hierarchy Level

[edit logical-systems *logical-system-name* protocols isis],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols isis],
[edit protocols isis],
[edit routing-instances *routing-instance-name* protocols isis]

Release Information

Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description

Configure alternate IS-IS topologies.

The remaining statements are explained separately.

Required Privilege Level

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

Related Topics

- Configuring IS-IS Multicast Topologies on page 338

traceoptions

Syntax

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

Hierarchy Level

[edit logical-systems *logical-system-name* protocols isis],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols isis],
[edit protocols isis],
[edit routing-instances *routing-instance-name* protocols isis]

Release Information

Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description

Configure IS-IS protocol-level tracing options. To specify more than one tracing operation, include multiple flag statements.

Default

The default IS-IS protocol-level tracing options are those inherited from the routing protocols traceoptions statement included at the [edit routing-options] hierarchy level.

Options

disable—(Optional) Disable the tracing operation. You can use this option to disable a single operation when you have defined a broad group of tracing operations, such as all.

file *name*—Name of the file to receive the output of the tracing operation. Enclose the name within quotation marks (" "). All files are placed in the directory /var/log. We recommend that you place IS-IS tracing output in the file isis-log.

files *number*—(Optional) Maximum number of trace files. When a trace file named *trace-file* reaches its maximum size, it is renamed *trace-file*.0, then *trace-file*.1, and so on, until the maximum number of trace files is reached. Then, the oldest trace file is overwritten.

If you specify a maximum number of files, you also must specify a maximum file size with the size option.
Range: 2 through 1000 files
Default: 10 files

flag *flag*—Tracing operation to perform. To specify more than one flag, include multiple flag statements.

IS-IS Protocol-Specific Tracing Flags

- csn—Complete sequence number PDU (CSNP) packets
- error—Errored IS-IS packets
- graceful-restart—Graceful restart operation
- hello—Hello packets

- ldp-synchronization—Synchronization between IS-IS and LDP
- lsp—Link-state PDU packets
- lsp-generation—Link-state PDU generation packets
- packets—All IS-IS protocol packets
- psn—Partial sequence number PDU (PSNP) packets
- spf—Shortest-path-first calculations

Global Tracing Flags

- all—All tracing operations
- general—A combination of the normal and route trace operations
- normal—All normal operations, including adjacency changes

  Default: If you do not specify this option, only unusual or abnormal operations are traced.
- policy—Policy operations and actions
- route—Routing table changes
- state—State transitions
- task—Routing protocol task processing
- timer—Routing protocol timer processing

flag-modifier—(Optional) Modifier for the tracing flag. You can specify one or more of these modifiers:

- detail—Detailed trace information
- receive—Packets being received
- send—Packets being transmitted

no-world-readable—(Optional) Prevent any user from reading the log file.

size *size*—(Optional) Maximum size of each trace file, in kilobytes (KB), megabytes (MB), or gigabytes (GB). When a trace file named *trace-file* reaches this size, it is renamed *trace-file*.0. When the *trace-file* again reaches its maximum size, *trace-file*.0 is renamed *trace-file*.1 and *trace-file* is renamed *trace-file*.0. This renaming scheme continues until the maximum number of trace files is reached. Then, the oldest trace file is overwritten.
Note that if you specify a maximum file size, you also must specify a maximum number of trace files with the files option.
Syntax: *x*k to specify KB, *x*m to specify MB, or *x*g to specify GB
Range: 10 KB through the maximum file size supported on your system
Default: 128 KB

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

Required Privilege Level routing and trace—To view this statement in the configuration.
routing-control and trace-control—To add this statement to the configuration.

Related Topics
- Tracing IS-IS Protocol Traffic on page 369

traffic-engineering

Syntax
```
traffic-engineering {
    credibility-protocol-preference;
    disable;
    family inet;
        shortcuts {
            multicast-rpf-routes;
        }
    }
    family inet6 {
        shortcuts;
    }
}
```

Hierarchy Level [edit logical-systems *logical-system-name* protocols isis],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols isis],
[edit protocols isis],
[edit routing-instances *routing-instance-name* protocols isis]

Release Information Statement introduced before JUNOS Release 7.4.
Support for the family statement introduced in JUNOS Release 9.3.
credibility-protocol-preference statement introduced in JUNOS Release 9.4.

Description Configure traffic engineering properties for IS-IS.

Default IS-IS traffic engineering support is enabled.

Options credibility-protocol-preference—Specify for IS-IS to use the configured protocol preference for IGP routes to determine the traffic engineering database credibility value. By default, the traffic engineering database prefers IS-IS routes even when the routes of another IGP are configured with a lower, that is, more preferred, preference value. Use this statement to override this default behavior.

The remaining statements are explained separately.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- traffic-engineering (OSPF)
- Configuring IS-IS Traffic Engineering Attributes on page 351

wide-metrics-only

Syntax wide-metrics-only;

Hierarchy Level [edit logical-systems *logical-system-name* protocols isis level *level-number*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols isis level *level-number*],
[edit protocols isis level *level-number*],
[edit routing-instances *routing-instance-name* protocols isis level *level-number*]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Configure IS-IS to generate metric values greater than 63 on a per IS-IS level basis.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- te-metric
- Enabling Wide IS-IS Metrics for Traffic Engineering on page 346

Chapter 17

Introduction to OSPF

This chapter discusses the following topics that provide background information about OSPF:

- OSPF Overview on page 432
- Understanding OSPF Areas on page 434
- Overview of Packets on page 436
- OSPF External Metrics Overview on page 439
- OSPF Designated Router Overview on page 439
- OSPF Extensions to Support Traffic Engineering on page 439
- OSPF Standards on page 440

OSPF Overview

OSPF is an IGP that routes packets within a single AS. OSPF uses link-state information to make routing decisions, making route calculations using the shortest path first (SPF) algorithm (also referred to as the Dijkstra algorithm). Each router running OSPF floods link-state advertisements throughout the AS that contain information about that router's attached interfaces and routing metrics. Each router takes the information in these link-state advertisements and creates a complete routing table for the network.

The JUNOS Software supports OSPF version 2, including virtual links, stub areas, and authentication. The JUNOS Software does not support type-of-service (ToS) routing.

OSPF was designed for the Transmission Control Protocol/IP (TCP/IP) environment and as a result explicitly supports IP subnetting and the tagging of externally derived routing information. OSPF also provides for the authentication of routing updates.

OSPF routes IP packets based solely on the destination IP address contained in the IP packet header. OSPF quickly detects topological changes, such as when router interfaces become unavailable, and calculates new loop-free routes quickly and with a minimum of routing overhead traffic.

Each interface running OSPF is assigned a cost, which is a unitless number based on factors such as throughput, round-trip time, and reliability, which are used to determine how easy or difficult it is to reach a destination. If two or more routes to a destination have the same cost, OSPF distributes traffic equally among the routes, a process that is called *load balancing*.

Each router maintains a database that describes the topology of the AS. Each OSPF router has an identical topological database so that all routers in the area have a consistent view of the network. All routers maintain summarized topologies of other areas within an AS. Each router distributes information about its local state by flooding link-state advertisements throughout the AS. When the AS topology changes, OSPF ensures that the contents of all routers' topological databases converge quickly.

All OSPF protocol exchanges can be authenticated. This means that only trusted routers can participate in the AS's routing. A variety of authentication schemes can be used; a single authentication scheme is configured for each area, which enables some areas to use stricter authentication than others.

Externally derived routing data (for example, routes learned from BGP) is passed transparently throughout the AS. This externally derived data is kept separate from the OSPF link-state data. Each external route can be tagged by the advertising router, enabling the passing of additional information between routers on the boundaries of the AS.

NOTE: By default, the JUNOS Software is compatible with RFC 1583, *OSPF Version 2*. In JUNOS Release 8.5 and later, you can disable compatibility with RFC 1583 by including the no-rfc-1583 statement. For more information, see "Disabling OSPFv2 Compatibility with RFC 1583" on page 453.

This section discusses the following topics:

- OSPF Routing Algorithm on page 433
- OSPF Version 3 on page 434

OSPF Routing Algorithm

OSPF uses the shortest path first (SPF) algorithm, also referred to as the Dijkstra algorithm, to determine the route to reach each destination. All routers in an area run this algorithm in parallel, storing the results in their individual topological databases. Routers with interfaces to multiple areas run multiple copies of the algorithm. This section provides a brief summary of how the SPF algorithm works.

When a router starts, it initializes OSPF and waits for indications from lower-level protocols that the router interfaces are functional. The router then uses the OSPF hello protocol to acquire neighbors, doing this by sending hello packets to its neighbors and receiving their hello packets.

On broadcast or nonbroadcast multiaccess networks (physical networks that support the attachment of more than two routers), the OSPF hello protocol elects a designated router for the network. This router is responsible for sending *link-state advertisements* that describe the network, which reduces the amount of network traffic and the size of the routers' topological databases.

The router then attempts to form *adjacencies* with some of its newly acquired neighbors. (On multiaccess networks, only the designated router and backup designated router form adjacencies with other routers.) Adjacencies determine the distribution of routing protocol packets: routing protocol packets are sent and received only on adjacencies, and topological database updates are sent only along adjacencies. When adjacencies have been established, pairs of adjacent routers synchronize their topological databases.

A router sends LSA packets to advertise its state periodically and when the router's state changes. These packets include information about the router's adjacencies, which allows detection of nonoperational routers.

Using a reliable algorithm, the router floods LSAs throughout the area, which ensures that all routers in an area have exactly the same topological database. Each router uses the information in its topological database to calculate a shortest-path tree, with itself as the root. The router then uses this tree to route network traffic.

The description of the SPF algorithm up to this point has explained how the algorithm works within a single area (*intra-area routing*). For internal routers to be able to route to destinations outside the area (*interarea routing*), the area border routers must inject additional routing information into the area. Because the area border routers are connected to the backbone, they have access to complete topological data about the backbone. They use this information to calculate paths to all destinations outside its area and then advertise these paths to the area's internal routers.

AS boundary routers flood information about external ASs throughout the AS, except to stub areas. Area border routers are responsible for advertising the paths to all AS boundary routers.

OSPF Version 3

OSPFv3 is a modified version of OSPF that supports IP version 6 (IPv6) addressing. OSPFv3 differs from OSPFv2 in the following ways:

- All neighbor ID information is based on a 32-bit router ID.
- The protocol runs per link rather than per subnet.
- Router and network link-state advertisements (LSAs) do not carry prefix information.
- Two new LSA types are included: link-LSA and intra-area-prefix-LSA.
- Flooding scopes are as follows:
  - Link-local
  - Area
  - AS
- Link-local addresses are used for all neighbor exchanges except virtual links.
- Authentication is removed; the IPv6 authentication header relies on the IP layer.
- The packet format has changed as follows:
  - Version number 2 is now version number 3.
  - The db option field has been expanded to 24 bits.
  - Authentication information has been removed.
  - Hello messages do not have address information.
  - Two new option bits are included: R and V6.
- Type 3 summary LSAs have been renamed *inter-area-prefix-LSAs*.
- Type 4 summary LSAs have been renamed *inter-area-router-LSAs*.

Understanding OSPF Areas

In OSPF, a single AS can be divided into smaller groups called *areas*. This reduces the number of link-state advertisements and other OSPF overhead traffic sent on the network, and it reduces the size of the topological database that each router must maintain.

This section discusses the following topics:

- Areas on page 435
- Area Border Routers on page 435
- Backbone Areas on page 435
- AS Boundary Routers on page 435
- Stub Areas on page 435

- Not-So-Stubby Areas on page 436
- Transit Areas on page 436

Areas

An *area* is a set of networks and hosts within an AS that have been administratively grouped together. We recommend that you configure an area as a collection of contiguous IP subnetted networks. Routers that are wholly within an area are called *internal routers*. All interfaces on internal routers are directly connected to networks within the area.

The topology of an area is hidden from the rest of the AS, thus significantly reducing routing traffic in the AS. Also, routing within the area is determined only by the area's topology, providing the area with some protection from bad routing data.

All routers within an area have identical topological databases.

Area Border Routers

Routers that belong to more than one area are called *area border routers*. They maintain a separate topological database for each area to which they are connected.

Backbone Areas

An OSPF *backbone area* consists of all networks in area ID 0.0.0.0, their attached routers, and all area border routers. The backbone itself does not have any area border routers. The backbone distributes routing information between areas. The backbone is simply another area, so the terminology and rules of areas apply: a router that is directly connected to the backbone is an internal router on the backbone, and the backbone's topology is hidden from the other areas in the AS.

The routers that make up the backbone must be physically contiguous. If they are not, you must configure *virtual links* to create the appearance of backbone connectivity. You can create virtual links between any two area border routers that have an interface to a common nonbackbone area. OSPF treats two routers joined by a virtual link as if they were connected to an unnumbered point-to-point network.

AS Boundary Routers

Routers that exchange routing information with routers in other ASs are called *AS boundary routers*. They advertise externally learned routes throughout the AS. Any router in the AS—an internal router, an area border router, or a backbone router—can be an AS boundary router.

Every router within the AS knows the path to the AS boundary routers.

Stub Areas

Stub areas are areas through which or into which AS external advertisements are not flooded. You might want to create stub areas when much of the topological database consists of AS external advertisements. Doing so reduces the size of the topological

databases and therefore the amount of memory required on the internal routers in the stub area.

When an area border router is configured for a stub area, the router automatically advertises a default route in place of the external routes that are not being advertised within the stub area so that routers in the stub area can reach destinations outside the area.

The following restrictions apply to stub areas: you cannot create a virtual link through a stub area, and a stub area cannot contain an AS boundary router.

Not-So-Stubby Areas

An OSPF stub area has no external routes in it, so you cannot redistribute from another protocol into a stub area. A not-so-stubby area (NSSA) allows external routes to be flooded within the area. These routes are then leaked into other areas. However, external routes from other areas still do not enter the NSSA.

Transit Areas

Transit areas are used to pass traffic from one adjacent area to the backbone (or to another area if the backbone is more than two hops away from an area). The traffic does not originate in, nor is it destined for, the transit area.

Overview of Packets

There are several types of link-state advertisement packets, which are discussed in “Link-State Advertisement Packet Types” on page 438.

This section contains the following topics:

- OSPF Packet Header on page 436
- Hello Packets on page 437
- Database Description Packets on page 437
- Link-State Request Packets on page 438
- Link-State Update Packets on page 438
- Link-State Acknowledgment Packets on page 438
- Link-State Advertisement Packet Types on page 438

OSPF Packet Header

All OSPF packets have a common 24-byte header that contains all information necessary to determine whether OSPF should accept the packet. The header consists of the following fields:

- Version number—The current OSPF version number. This can be either 2 or 3.
- Type—Type of OSPF packet.
- Packet length—Length of the packet, in bytes, including the header.

- Router ID—IP address of the router from which the packet originated.
- Area ID—Identifier of the area in which the packet is traveling. Each OSPF packet is associated with a single area. Packets traveling over a virtual link are labeled with the backbone area ID, 0.0.0.0. You configure the area ID as described in "Configuring OSPF Areas" on page 449.
- Checksum—Fletcher checksum.
- Authentication—Authentication scheme and authentication information. You configure authentication as described in "Configuring Authentication for OSPFv2" on page 458.

Hello Packets

Routers periodically send hello packets on all interfaces, including virtual links, to establish and maintain neighbor relationships. Hello packets are multicast on physical networks that have a multicast or broadcast capability, which enables dynamic discovery of neighboring routers. (On nonbroadcast networks, dynamic neighbor discovery is not possible, so you must configure all neighbors statically as described in "Configuring an Interface on a Nonbroadcast, Multiaccess Network" on page 455.)

Hello packets consist of the OSPF header plus the following fields:

- Network mask—Network mask associated with the interface.
- Hello interval—How often the router sends hello packets. All routers on a shared network must use the same hello interval. You configure this interval as described in "Modifying the Hello Interval" on page 466.
- Options—Optional capabilities of the router.
- Router priority—The router's priority to become the designated router. You can configure this value as described in "Configuring the Designated Router Priority for OSPF" on page 462.
- Router dead interval—How long the router waits without receiving any OSPF packets from a router before declaring that router to be down. All routers on a shared network must use the same router dead interval. You can configure this value as described in "Modifying the Router Dead Interval" on page 467.
- Designated router—IP address of the designated router.
- Backup designated router—IP address of the backup designated router.
- Neighbor—IP addresses of the routers from which valid hello packets have been received within the time specified by the router dead interval.

Database Description Packets

When initializing an adjacency, OSPF exchanges database description packets, which describe the contents of the topological database. These packets consist of the OSPF header, packet sequence number, and the link-state advertisement's header.

Link-State Request Packets

When a router detects that portions of its topological database are out of date, it sends a link-state request packet to a neighbor requesting a precise instance of the database. These packets consist of the OSPF header plus fields that uniquely identify the database information that the router is seeking.

Link-State Update Packets

Link-state update packets carry one or more link-state advertisements one hop farther from their origin. The router multicasts (floods) these packets on physical networks that support multicast or broadcast mode. The router acknowledges all link-state update packets and, if retransmission is necessary, sends the retransmitted advertisements unicast.

Link-state update packets consist of the OSPF header plus the following fields:

- Number of advertisements—Number of link-state advertisements included in this packet.
- Link-state advertisements—The link-state advertisements themselves.

Link-State Acknowledgment Packets

The router sends link-state acknowledgment packets in response to link-state update packets to verify that the update packets have been received successfully. A single acknowledgment packet can include responses to multiple update packets.

Link-state acknowledgment packets consist of the OSPF header plus the link-state advertisement header.

Link-State Advertisement Packet Types

Link-state request, link-state update, and link-state acknowledgment packets are used to reliably flood link-state advertisement packets. OSPF sends the following types of link-state advertisements:

- Router link advertisements—Are sent by all routers to describe the state and cost of the router's links to the area. These link-state advertisements are flooded throughout a single area only.
- Network link advertisements—Are sent by designated routers to describe all the routers attached to the network. These link-state advertisements are flooded throughout a single area only.
- Summary link advertisements—Are sent by area border routers to describe the routes that they know about in other areas. There are two types of summary link advertisements: those used when the destination is an IP network, and those used when the destination is an AS boundary router. Summary link advertisements describe interarea routes; that is, routes to destinations outside

the area but within the AS. These link-state advertisements are flooded throughout the advertisement's associated areas.

- AS external link advertisement—Are sent by AS boundary routers to describe external routes that they know about. These link-state advertisements are flooded throughout the AS (except for stub areas).

Each link-state advertisement type describes a portion of the OSPF routing domain. All link-state advertisements are flooded throughout the AS.

Each link-state advertisement packet begins with a common 20-byte header.

OSPF External Metrics Overview

When OSPF exports route information from external ASs, it includes a cost, or *external metric*, in the route. There are two types of external metrics: Type 1 and Type 2. Type 1 external metrics are equivalent to the link-state metric; that is, the cost of the route used in the internal AS. Type 2 external metrics are greater than the cost of any path internal to the AS.

OSPF Designated Router Overview

Each multiaccess network has a designated router, which performs two main functions:

- Originate network link advertisements on behalf of the network.
- Establish adjacencies with all routing devices on the network, thus participating in the synchronizing of the link-state databases.

The OSPF hello protocol elects a designated router for the network based on the priorities advertised by all the routing devices. In general, when an interface first becomes functional, it checks whether the network currently has a designated router. If there is one, the routing device accepts that designated router regardless of its own router priority. Otherwise, if the routing device has the highest priority on the network, it becomes the designated router. If router priorities tie, the routing device with the highest router ID (which is typically the routing device's IP address) is chosen as the designated router.

OSPF Extensions to Support Traffic Engineering

To help provide traffic engineering and MPLS with information about network topology and loading, extensions have been added to the JUNOS Software implementation of OSPF. Specifically, OSPF generates opaque (link-state advertisements (LSAs), which carry traffic engineering parameters. These parameters are used to populate the traffic engineering database, which is used by the Constrained Shortest Path First (CSPF) algorithm to compute the paths that MPLS LSPs take. This path information is used by RSVP to set up LSPs and reserve bandwidth for them.

This section also includes:

Configuring OSPF IGP Shortcuts

In OSPF, you can configure shortcuts, which allow OSPF to use an LSP as the next hop as if it were a logical interface from the ingress routing device to the egress routing device. The address specified on the to statement at the [edit protocols mpls label-switched-path *lsp-path-name*] hierarchy level must match the router ID of the egress routing device for the LSP to function as a direct link to the egress routing device and to be used as input to OSPF SPF calculations. When used in this way, LSPs are no different than Asynchronous Transfer Mode (ATM) and Frame Relay virtual circuits (VCs), except that LSPs carry only IPv4 traffic.

NOTE: Whenever possible, use OSPF IGP shortcuts instead of traffic engineering shortcuts.

OSPF Standards

OSPF and OSPFv3 are defined in the following documents:

- RFC 1793, *Extending OSPF to Support Demand Circuits*
- RFC 2328, *OSPF Version 2*
- RFC 2370, *The OSPF Opaque LSA Option*
- RFC 2740, *OSPF for IPv6*
- RFC 3101, *The OSPF Not-So-Stubby Area (NSSA) Option*
- RFC 3509, *Alternative Implementations of OSPF Area Border Routers*
- RFC 3623, *OSPF Graceful Restart*
- RFC 3630, *Traffic Engineering (TE) Extensions to OSPF Version 2*
- RFC 4203, *OSPF Extensions in Support of Generalized Multi-Protocol Label Switching (GMPLS) (only interface switching)*
- RFC 4552, *Authentication/Confidentiality for OSPFv3*
- RFC 4576, *Using a Link State Advertisement (LSA) Options Bit to Prevent Looping in BGP/MPLS Virtual Private Networks (VPNs)*
- RFC 4577, *OSPF as the Provider/Customer Edge Protocol for BGP/MPLS IP Virtual Private Networks (VPNs)*
- RFC 5185, *OSPF Multi-Area Adjacency*
- Internet draft draft-ietf-katz-ward-bfd-00.txt, *Bidirectional Forwarding Detection* (except the transmission of echo packets) (expires January 2005)
- Internet draft draft-ietf-isis-igp-p2p-over-lan-03.txt, *Point-to-point operation over LAN in link-state routing protocols* (expires February 2004)
- Internet draft draft-ospf-alt-06.txt, *Support of address families in OSPFv3* (expires April 2008)

To access Internet RFCs and drafts, go to the IETF website at http://www.ietf.org.

Chapter 18

OSPF Configuration Guidelines

This chapter describes the following tasks for configuring OSPF:

- Configuring OSPF on page 444
- Minimum OSPF Configuration on page 449
- Configuring OSPF Areas on page 449
- Disabling Export of LSAs into NSSAs Attached to ASBR ABRs on page 453
- Disabling OSPFv2 Compatibility with RFC 1583 on page 453
- Configuring OSPF on Interfaces on page 454
- Configuring Multiarea Adjacency in OSPFv2 on page 457
- Configuring Multiple Address Families for OSPFv3 on page 457
- Configuring Authentication for OSPFv2 on page 458
- Configuring Authentication for OSPFv3 on page 462
- Limiting the Number of Prefixes Exported to OSPF on page 462
- Configuring the Designated Router Priority for OSPF on page 462
- Summarizing Ranges of Routes in OSPF Link-State Advertisements on page 463
- Configuring the Metric Value for OSPF Interfaces on page 464
- Dynamically Adjusting OSPF Interface Metrics Based on Bandwidth on page 465
- Configuring Preference Values for OSPF Routes on page 465
- Configuring OSPF Timers on page 466
- Configuring OSPF Refresh and Flooding Reduction in Stable Topologies on page 468
- Configuring BFD for OSPF on page 469
- Overview of BFD Authentication for OSPF on page 471
- Configuring BFD Authentication for OSPF on page 473
- Configuring Synchronization Between LDP and IGPs on page 476
- Configuring Graceful Restart for OSPF and OSPFv3 on page 477
- Configuring SPF Options for OSPF on page 478
- Advertising Interface Addresses Without Running OSPF on page 479
- Configuring OSPF Passive Traffic Engineering Mode on page 479
- Advertising Label-Switched Paths into OSPF on page 480

- Configuring OSPF to Make Routing Devices Appear Overloaded on page 481
- Enabling OSPF Traffic Engineering Support on page 481
- Configuring the OSPF Metric Value Used for Traffic Engineering on page 484
- Applying Policies to OSPF Routes on page 484
- Configuring OSPF Routing Table Groups on page 487
- Configuring OSPF Sham Links on page 488
- Configuring OSPF Peer Interfaces on page 488
- Loop-Free Alternate Routes for OSPF Overview on page 489
- Configuring Link Protection for OSPF on page 490
- Configuring Node-Link Protection for OSPF on page 491
- Excluding an OSPF Interface as a Backup for a Protected Interface on page 492
- Configuring Backup SPF Options for Protected OSPF Interfaces on page 493
- Configuring RSVP Label-Switched Paths as Backup Paths for OSPF on page 495
- Tracing OSPF Protocol Traffic on page 495

Configuring OSPF

To configure OSPF version 2 (usually referred to simply as OSPF), you include the following statements:

```
protocols {
    ospf {
        disable;
        export [ policy-names ];
        external-preference preference;
        graceful-restart {
            disable;
            helper-disable;
            notify-duration seconds;
            restart-duration seconds;
        }
        import [ policy-names ];
        no-nssa-abr;
        no-rfc-1583;
        overload {
            timeout seconds;
        }
        preference preference;
        prefix-export-limit;
        rib-group group-name;
        reference-bandwidth reference-bandwidth;
        sham-link {
            local address;
        }
        spf-options {
            delay milliseconds;
            rapid-runs number;
            holddown milliseconds;
```

```
}
traffic-engineering {
   advertise-unnumbered-interfaces;
   multicast-rpf-routes;
   no-topology;
   shortcuts {
      ignore-lsp-metrics;
      lsp-metric-into-summary;
   }
}
traceoptions {
   file filename <files number> <size size> <world-readable | no-world-readable>;
   flag flag <flag-modifier> <disable>;
}
area area-id {
   area-range network/mask-length <restrict> <exact> <override-metric metric>;
   interface interface-name {
      disable;
      authentication {
         md5 key-id {
            key [ key-values ];
            start-time time;
         }
         simple-password key;
      }
      bfd-liveness-detection {
         authentication {
            algorithm algorithm-name;
            key-chain key-chain-name;
            loose-check;
         }
         detection-time {
            threshold milliseconds;
         }
         full-neighbors-only;
         minimum-interval milliseconds;
         minimum-receive-interval milliseconds;
         multiplier number;
         no-adaptation;
         transmit-interval {
            threshold milliseconds;
            minimum-interval milliseconds;
         }
         version (1 | automatic);
      }
      dead-interval seconds;
      demand-circuit;
      flood-reduction;
      hello-interval seconds;
      interface-type type;
      ipsec-sa name;
      ldp-synchronization {
         disable;
         hold-time seconds;
      }
      metric metric;
```

```
        neighbor address <eligible>;
        passive {
            traffic-engineering {
                remote-node-id address;
            }
        }
        poll-interval seconds;
        priority number;
        retransmit-interval seconds;
        secondary;
        te-metric metric;
        topology (ipv4-multicast | name) {
            metric metric;
        }
        transit-delay seconds;
    }
    label-switched-path name metric metric;
    network-summary-export [ policy-names ];
    network-summary-import [policy-names ];
    nssa {
        area-range network/mask-length <restrict> <exact> <override-metric metric>;
        default-lsa {
            default-metric metric;
            metric-type type;
            type-7;
        }
        (summaries | no-summaries);
    }
    peer-interface interface-name {
        disable;
        dead-interval seconds;
        demand-circuit;
        flood-reduction;
        hello-interval seconds;
        retransmit-interval seconds;
        transit-delay seconds;
    }
    sham-link-remote address {
        demand-circuit;
        flood-reduction;
        ipsec-sa name;
        metric metric;
    }
}
stub <default-metric metric> <summaries | no-summaries>;
virtual-link neighbor-id router-id transit-area area-id {
    disable;
    authentication {
        md5 key-id {
            key [ key-values ];
        }
        simple-password key;
    }
    dead-interval seconds;
    demand-circuit;
    flood-reduction;
```

```
        hello-interval seconds;
        ipsec-sa name;
        retransmit-interval seconds;
        topology (ipv4-multicast | name) disable;
        transit-delay seconds;
      }
    }
  }
```

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for these statements.

To configure OSPF version 3 (OSPFv3), you include the following statements:

```
protocols {
  ospf3 {
    disable;
    export [ policy-names ];
    external-preference preference;
    graceful-restart {
      disable;
      helper-disable;
      notify-duration seconds;
      restart-duration seconds;
    }
    import [ policy-names ];
    overload {
      timeout seconds;
    }
    preference preference;
    prefix-export-limit;
    reference-bandwidth reference-bandwidth;
    realm (ipv4-unicast | ipv4-multicast | ipv6-multicast);
    rib-group group-name;
    spf-options {
      delay milliseconds;
      holddown milliseconds;
      rapid-runs number;
    }
    traceoptions {
      file filename <files number> <size size> <world-readable | no-world-readable>;
      flag flag <flag-modifier> <disable>;
    }
    area area-id {
      area-range network/mask-length <restrict> <exact> <override-metric metric>;
      interface interface-name {
        bfd-liveness-detection {
          detection-time {
            threshold milliseconds;
          }
          full-neighbors-only;
          minimum-interval milliseconds;
          minimum-receive-interval milliseconds;
          multiplier number;
          no-adaptation;
          transmit-interval {
```

```
                threshold milliseconds;
                minimum-interval milliseconds;
            }
            version (1 | automatic);
        }
        disable;
        dead-interval seconds;
        flood-reduction;
        hello-interval seconds;
        ipsec-sa name;
        metric metric;
        passive {
            traffic-engineering {
                remote-node-id address;
            }
        }
        priority number;
        retransmit-interval seconds;
        transit-delay seconds;
    }
    inter-area-prefix-export policy-name;
    inter-area-prefix-import policy-name;
    nssa {
        area-range network/mask-length <restrict> <exact> <override-metric metric>;
        default-lsa {
            default-metric metric;
            metric-type type;
            type-7;
        }
        (summaries | no-summaries);
    }
    stub <default-metric metric> <summaries | no-summaries>;
    virtual-link neighbor-id router-id transit-area area-id {
        disable;
        dead-interval seconds;
        hello-interval seconds;
        flood-reduction;
        ipsec-sa name;
        retransmit-interval seconds;
        transit-delay seconds;
    }
  }
 }
}
```

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for these statements.

For a detailed OSPFv3 example configuration, see the *JUNOS Feature Guide*.

By default, OSPFv2 and OSPFv3 are disabled.

NOTE: In this manual, the term *OSPF* refers to both OSPFv2 and OSPFv3.

Minimum OSPF Configuration

You must create a backbone area if your network consists of multiple areas. An area border router (ABR) must have at least one interface in the backbone area, or it must have a virtual link to a routing device in the backbone area. To do this, include at least the following statements. All other OSPF configuration statements are optional.

```
protocols {
   (ospf | ospf3 ) {
      area 0 {
         interface interface-name;
      }
   }
}
```

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for these statements.

NOTE: When you configure OSPFv2 on an interface, you must also include the family inet statement at the [edit interfaces *interface-name* unit *logical-unit-number*] hierarchy level. When you configure OSPFv3 on an interface, you must also include the family inet6 statement at the [edit interfaces *interface-name* unit *logical-unit-number*] hierarchy level. For more information about the family inet statement, see the *JUNOS Network Interfaces Configuration Guide*.

NOTE: OSPFv3 does not support routing instances.

Configuring OSPF Areas

You can group the routing devices in a single autonomous system (AS) into areas to reduce the amount of link-state advertisement (LSA) traffic on the network and to reduce the size of the topological databases that OSPF routing devices must maintain. If you do this, the AS must contain a single backbone area and optionally can contain any number of nonbackbone areas. The routing devices that make up the backbone must be physically contiguous. If they are not, you must configure virtual links to create the appearance of connectivity. You also can configure stub areas, which are areas through which AS external advertisements are not flooded, and not-so-stubby areas (NSSAs), which allow external routes to be flooded within an area.

The JUNOS Software supports active backbone detection. Active backbone detection is implemented to verify that area border routers are connected to the backbone. If the connection to the backbone area is lost, then the routing device's default metric is not advertised, effectively rerouting traffic through another area border router with a valid connection to the backbone.

Active backbone detection enables transit through an area border router (ABR) with no active backbone connection. An ABR advertises to other routing devices that it is

an ABR even if the connection to the backbone is down, so that the neighbors can consider it for interarea routes.

To configure areas, you can perform the following tasks:

- Configuring the OSPF Backbone Area on page 450
- Configuring OSPF Nonbackbone Areas on page 450
- Configuring OSPF Stub Areas on page 450
- Configuring OSPF Not-So-Stubby Areas on page 451
- Configuring OSPF Virtual Links on page 452

Configuring the OSPF Backbone Area

You must create a backbone area if your network consists of multiple areas. An ABR must have at least one interface in the backbone area, or it must have a virtual link to a routing device in the backbone area. The backbone comprises all ABRs and all routing devices that are not included in any other area. You configure all these routing devices by including the area 0.0.0.0 statement:

```
area 0.0.0.0;
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for the statement.

Configuring OSPF Nonbackbone Areas

Each OSPF area consists of routing devices configured with the same area number. To configure a routing device to be in an area, include the area statement. The area ID can be any number except 0.0.0.0, which is reserved for the backbone area.

```
area area-id;
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for the statement.

Configuring OSPF Stub Areas

Stub areas are areas into which OSPF does not flood AS external advertisements. You might want to configure stub areas when much of the topological database consists of AS external advertisements and you want to minimize the size of the topological databases on an area's routing devices.

You cannot configure an area as being both a stub area and an not-so-stubby area (NSSA).

To configure a stub area, include the stub statement:

```
area area-id {
    stub <default-metric metric> <summaries | no-summaries>;
}
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

To inject a default route with a specified metric value into the area, include the default-metric option and a metric value. The default route matches any destination that is not explicitly reachable from within the area.

To have the stub areas not advertise summary routes into the stub area, include the no-summaries option. Only the default route is advertised, and only if you include the default-metric option. The default route injected into the NSSA is a Type 3 LSA.

You must include the stub statement when configuring all routing devices that are in the stub area.

NOTE: In JUNOS Release 8.5 and later, a router-identifier interface that is not configured to run OSPF is no longer advertised as a stub network in OSPF link-state advertisements. For more information about how to configure a router identifier, see "Configuring Router Identifiers for BGP and OSPF" on page 113.

In JUNOS Release 8.5 and later, OSPF advertises a local route with a prefix length of 32 as a stub link if the loopback interface is configured with a prefix length other than 32. OSPF also advertises the direct route with the configured mask length, as in earlier releases.

Configuring OSPF Not-So-Stubby Areas

An OSPF stub area has no external routes, so you cannot redistribute from another protocol into a stub area. An NSSA allows external routes to be flooded within the area. These routes are then leaked into other areas. However, external routes from other areas still do not enter the NSSA.

You cannot configure an area to be both a stub area and an NSSA.

To configure an NSSA, include the nssa statement:

```
area area-id {
    nssa {
        area-range network/mask-length <restrict> <exact> <override-metric metric>;
        default-lsa {
            default-metric metric;
            metric-type type;
            type-7;
        }
        (summaries | no-summaries);
    }
}
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

By default, a default route is not advertised. To advertise a default route with the specified metric within the area, include the default-metric statement. You can configure this option only on ABRs.

To prevent an ABR from advertising summary routes into an NSSA, include the no-summaries statement. If you include the default-metric option in addition to the no-summaries statement, only the default route is advertised. The default route is a Type 3 LSA injected into the NSSA. To flood summary LSAs into the NSSA area, include the summaries statement. When summaries is configured (which is the default if the no-summaries statement is not specified), a Type 7 LSA is sent. To define the type of metric, include the metric-type statement.

To aggregate external routes learned within the area when a route is advertised to other areas, include one or more area-range statements. If you also include the restrict option, the aggregate is not advertised, effectively creating a route filter. All external routes learned within the area that do not fall into the range of one of the prefixes are advertised individually to other areas. To restrict an exact area range, include the exact option. For an example, you can suppress the exact 0/0 prefix from being advertised from a NSSA area into the backbone area by including both the exact and restrict options. To override the metric for the IP address range and configure a specific metric value, include the override-metric option.

Configuring OSPF Virtual Links

If any routing device on the backbone is not physically connected to the backbone itself, you must establish a virtual connection between that routing device and the backbone. You can establish a virtual connection between ABRs by configuring an OSPF virtual link.

To configure an OSPF virtual link, include the virtual-link statement when configuring the backbone area (area 0):

```
virtual-link neighbor-id router-id transit-area area-id;
```

To configure an OSPFv3 virtual link, include the virtual-link statement when configuring the backbone area (area 0):

```
virtual-link neighbor-id router-id transit-area area-id;
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

Specify the router ID (as an IPv4 address) of the routing device at the other end of the virtual link. This router must be an ABR that is physically connected to the backbone. Also, specify the number of the area through which the virtual link transits.

For the virtual connection to work, you also must configure a link to the backbone area on the remote ABR (the routing device at the other end of the LSP).

Example: Configuring an OSPF Virtual Link

Configure an OSPF virtual link on the local router. This routing device must be an area border router that is physically connected to the backbone.

```
[edit protocols ospf]
area 0.0.0.0 {
   virtual-link neighbor-id 192.168.0.3 transit-area 1.1.1.1;
   interface t3-1/0/0 {
      hello-interval 1;
      dead-interval 3;
   }
}
```

You must also configure an OSPF virtual link on the remote ABR:

```
[edit protocols ospf]
area 0.0.0.0 {
   virtual-link neighbor-id 192.168.0.5 transit-area 1.1.1.1;
}
```

Disabling Export of LSAs into NSSAs Attached to ASBR ABRs

When an autonomous-system boundary router (ASBR) is also an ABR with an NSSA area attached to it, a Type 7 LSA is exported into the NSSA area by default. If the ABR is attached to multiple NSSA areas, a separate Type 7 LSA is exported into each NSSA area by default.

NOTE: Type 7 LSAs are not exported into an NSSA if there is only one NSSA and backbone area connected to the ABR.

To disable exporting Type 7 LSAs into NSSAs, include the no-nssa-abr statement:

```
no-nssa-abr;
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

Disabling OSPFv2 Compatibility with RFC 1583

By default, the JUNOS implementation of OSPFv2 is compatible with RFC 1583, *OSPF Version 2*. This means that the JUNOS Software maintains a single best route to an AS boundary router in the OSPF routing table, rather than multiple intra-AS paths, if they are available. You can now disable compatibility with RFC 1583. It is preferable to do so when the same external destination is advertised by AS boundary routers that belong to different OSPF areas. When you disable compatibility with RFC 1583, the OSPF routing table maintains the multiple intra-AS paths that are available, which the router uses to calculate AS external routes as defined in RFC 2328, *OSPF Version 2*. Being able to use multiple, available paths to calculate an AS external route can prevent routing loops.

To disable OSPF v2 compatibility with RFC 1583, include the no-rfc-1583 statement:

```
no-rfc-1583;
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

Configuring OSPF on Interfaces

To enable OSPF on the routing device, you must configure OSPF on at least one of the routing device's interfaces. How you configure an interface depends on whether the interface is connected to a broadcast or point-to-point network, a point-to-multipoint network, or a nonbroadcast, multiaccess network.

NOTE: When you configure OSPFv2 on an interface, you must also include the family inet statement at the [edit interfaces *interface-name* unit *logical-unit-number*] hierarchy level. When you configure OSPFv3 on an interface, you must also include the family inet6 statement at the [edit interfaces *interface-name* unit *logical-unit-number*] hierarchy level. For more information about the family inet statement, see the *JUNOS Network Interfaces Configuration Guide*. In JUNOS Release 9.2 and later, you can configure OSPFv3 to support address families other than unicast IPv6. For more information, see "Configuring Multiple Address Families for OSPFv3" on page 457.

To configure OSPF on an interface, you can perform the following tasks:

- Configuring an Interface on a Broadcast or Point-to-Point Network on page 454
- Configuring an Interface on a Point-to-Multipoint Network on page 455
- Configuring an Interface on a Nonbroadcast, Multiaccess Network on page 455
- Configuring an OSPF Demand Circuit Interface on page 456

Configuring an Interface on a Broadcast or Point-to-Point Network

If the interface on which you are configuring OSPF supports broadcast mode (such as a LAN), or if the interface supports point-to-point mode (such as a PPP interface or a point-to-point logical interface on Frame Relay), include the following form of the interface statement:

```
area area-id {
    interface interface-name;
}
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

Specify the interface by IP address or interface name for OSPFv2, or only the interface name for OSPFv3. In JUNOS Release 9.3 and later, an OSPF point-to-point interface can be an Ethernet interface without a subnet. For more information about interface names, see the *JUNOS Network Interfaces Configuration Guide*.

NOTE: Using both the interface name and IP address of the same interface produces an invalid configuration.

Configuring an Interface on a Point-to-Multipoint Network

When you configure OSPFv2 on a nonbroadcast multiaccess (NBMA) network, such as a multipoint ATM or Frame Relay, OSPFv2 operates by default in point-to-multipoint mode. In this mode, OSPFv2 treats the network as a set of point-to-point links. Because there is no autodiscovery mechanism, each neighbor must be configured.

To configure OSPFv2 in point-to-multipoint mode, include the following statement:

```
interface interface-name {
    neighbor address;
}
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

Specify the interface by IP address or interface name. Using both the IP address and interface name produces an invalid configuration. For more information about interface names, see the *JUNOS Network Interfaces Configuration Guide*.

To configure multiple neighbors, include a neighbor statement for each neighbor.

Configuring an Interface on a Nonbroadcast, Multiaccess Network

When configuring OSPFv2 on an NBMA network, you can use nonbroadcast mode rather than point-to-multipoint mode. Using this mode offers no advantages over point-to-multipoint mode, but it has more disadvantages than point-to-multipoint mode. Nevertheless, you might occasionally find it necessary to configure nonbroadcast mode to interoperate with other equipment.

Nonbroadcast mode treats the NBMA network as a partially connected LAN, electing designated and backup designated routers. All routing devices must have a direct connection to both the designated and backup designated routers, or unpredictable results occur.

To configure nonbroadcast mode, include the following statements:

```
interface interface-name {
    interface-type nbma;
    neighbor address <eligible>;
    poll-interval seconds;
}
```

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for these statements.

Specify the interface by IP address or interface name. Using both an IP address and interface name produces an invalid configuration. For more information about interface names, see the *JUNOS Network Interfaces Configuration Guide*.

NOTE: For nonbroadcast interfaces, specify the IP address of the nonbroadcast interface as the *interface-name*.

To configure multiple neighbors, include a neighbor statement for each neighbor.

OSPF routing devices normally discover their neighbors dynamically by listening to the broadcast or multicast hello packets on the network. Because an NBMA network does not support broadcast (or multicast), the routing device cannot discover its neighbors dynamically, so you must configure all the neighbors statically. Do this by including the neighbor statement and specifying the IP address of each neighboring routing device in the *address* option. To configure multiple neighbors, include multiple neighbor statements. If the neighbor is allowed to become the designated router, include the eligible keyword.

By default, the routing device sends hello packets out the interface every 120 seconds before it establishes adjacency with a neighbor. To modify this interval, include the poll-interval statement.

Configuring an OSPF Demand Circuit Interface

A demand circuit is a connection on which you can limit traffic based on user agreements. The demand circuit can limit bandwidth or access time based upon agreements between the provider and user.

Demand circuits can be used to implement Integrated Services Digital Network (ISDN). For this application, demand circuits are configured on point-to-point and point-to-multipoint interfaces. For more information on ISDN, see the *JUNOS Software Interfaces and Routing Configuration Guide*.

Demand circuits can be configured on an OSPF interface. When the interface becomes a demand circuit, all hello packets and link-state advertisements are suppressed as soon as OSPF synchronization is achieved. Hello packets and link-state advertisements are sent and received on a demand-circuit interface only when there is a change in the network topology. This reduces the amount of traffic through the OSPF interface.

To configure an OSPF interface as a demand circuit, include the demand-circuit statement:

```
area area-id {
    demand-circuit;
}
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

A demand-circuit interface automatically negotiates demand-circuit connection with its OSPF neighbor. If the neighbor does not support demand circuits, then no demand circuit connection is established.

Configuring Multiarea Adjacency in OSPFv2

By default, a single interface can belong to only one OSPF area. However, in some situations, you might want to configure an interface to belong to more than one area. Doing so allows the corresponding link to be considered an intra-area link in multiple areas and to be preferred over other higher-cost intra-area paths. For example, you configure an interface to belong to multiple areas with a high-speed backbone link between two area border routers to enable you to create multiarea adjacencies that belong to different areas.

In JUNOS Release 9.2 and later, you can configure a logical interface to belong to more than one OSPF area. As defined in RFC 5185, *OSPF Multi-Area Adjacency*, the area border routers establish multiple adjacencies belonging to different areas over the same logical interface. Each multiarea adjacency is announced as a point-to-point unnumbered link in the configured area by the routers connected to the link. For each area, one of the logical interfaces is treated as primary, and the remaining interfaces that are configured for the area are designated as secondary.

To configure a secondary logical interface for an OSPF area, include the secondary statement:

```
area area-id {
    interface interface-name {
        secondary;
    }
}
```

Any logical interface not configured as a secondary interface for an area is treated as a primary interface for that area. A logical interface can be configured as primary interface only for one area. For any other area for which you configure the interface, you must configure it as a secondary interface.

NOTE: You cannot configure the secondary statement with the interface all statement. You also cannot configure as secondary an interface by its IP address.

For a list of hierarchy levels at which you can include the statement, see the statement summary section for this statement.

Configuring Multiple Address Families for OSPFv3

By default, OSPFv3 supports only unicast IPv6 routes. In JUNOS Release 9.2 and later, you can configure OSPFv3 to support multiple address families, including IPv4 unicast, IPv4 multicast, and IPv6 multicast. The JUNOS Software maps each address family to a separate realm as defined in Internet draft draft-ietf-ospf-af-alt-06.txt, *Support for Address Families in OSPFv3*. Each realm maintains a separate set of neighbors and link-state database.

To configure an OSPFv3 realm, include the realm (ipv4-unicast | ip4-multcast | ipv6-multicast) statement:

```
realm (ipv4-unicast | ipv4-multicast | ipv6-multicast)
   area area-id {
      interface interface-name;
   }
}
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

You configure each realm independently. We recommend that you configure an area and at least one interface for each realm.

These are the default import and export routing tables for each of the four address families:

- IPv6 unicast: inet6.0
- IPv6 multicast: inet6.2
- IPv4 unicast: inet.0
- IPv4 multicast: inet.2

With the exception of virtual links, all configurations supported for the default IPv6 unicast family are supported for the address families that have to be configured as realms.

Configuring Authentication for OSPFv2

All OSPFv2 protocol exchanges can be authenticated to guarantee that only trusted routers participate in the autonomous system's routing. By default, OSPFv2 authentication is disabled. JUNOS Software supports MD5 and simple authentication, and in JUNOS Release 8.3 and later, IPsec authentication. You can configure IPsec authentication for the OSPFv2 interface, the remote endpoint of a sham link, and the OSPFv2 virtual link.

NOTE: You can configure IPsec authentication together with either MD5 or simple authentication.

- To enable IPsec authentication for an OSPFv2 interface, include the ipsec-sa *name* statement for a specific interface:

  interface *interface-name* ipsec-sa *name*;

- To enable IPsec authentication for a remote sham link, include the ispec-sa *name* statement for the remote end point of the sham link:

  sham-link-remote *address* ipsec-sa *name*;

NOTE: If a Layer 3 VPN configuration has multiple sham links with the same remote endpoint IP address, you must configure the same IPsec security association for all the remote endpoints. You configure a Layer 3 VPN at the [edit routing-instances *routing-instance-name* instance-type] hierarchy level. For more information about Layer 3 VPNs, see the *JUNOS VPNs Configuration Guide*.

- To enable IPsec authentication for a virtual link, include the ipsec-sa *name* statement for a specific virtual link:

  virtual-link neighbor-id *router-id* transit-area *area-id* ipsec-sa *name*;

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for these statements.

You specify the IPsec authentication name by including the ipsec-sa *name* statement where *name* is the name of the IPsec security association. You configure the actual IPsec authentication separately. Only manual security associations (SAs) are supported for OSPFv2 authentication using IPsec. Dynamic IKE SAs are not supported. For more information about IPsec, see the *JUNOS System Basics Configuration Guide*, the *JUNOS Services Interfaces Configuration Guide*, and the *JUNOS Feature Guide*.

The following restrictions also apply to IPsec authentication for OSPFv2:

- Only IPsec transport mode is supported. Tunnel mode is not supported.
- Because only bidirectional manual SAs are supported, all OSPFv2 peers must be configured with the same IPsec SA. You configure a manual bidirectional SA at the [edit security ipsec] hierarchy level.
- You must also configure the same IPsec SA for all virtual links with the same remote endpoint address, for all neighbors on OSPF nonbroadcast multiaccess (NBMA) or point-to-multipoint (P2MP) links, and for every subnet that is part of a broadcast link.
- OSPFv2 peer interfaces are not supported.

Simple authentication uses a text password that is included in the transmitted packet. The receiving router uses an authentication key (password) to verify the packet.

The MD5 algorithm creates an encoded checksum that is included in the transmitted packet. The receiving router uses an authentication key (password) to verify the packet.

For MD5 authentication to work, both the receiving and transmitting routers must have the same MD5 key. Define an MD5 key for each interface. If MD5 is enabled on an interface, that interface accepts routing updates only if MD5 authentication succeeds; otherwise, updates are rejected. The key ID can be set to any value between 0 and 255, with a default value of 0. The router only accepts OSPFv2 packets sent using the same key ID that is defined for that interface.

To enable authentication and specify an authentication method as well as a key (password) for an OSPF interface or virtual link, include the authentication statement and either a single simple-password statement or one or more md5 statements:

```
authentication {
    simple-password key;
}
authentication {
    md5 key {
        key [ key-values ] {
        start-time time;
    }
}
```

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for these statements.

A simple password and MD5 key are mutually exclusive. You can configure only one simple password but configure multiple MD5 keys.

The simple key (password) can be from 1 through 8 characters long. Each MD5 key is identified by a key identifier. The MD5 key value can be from 1 through 16 characters long. Characters can include ASCII strings. If you include spaces, enclose all characters in quotation marks (“ ”).

As part of your security measures, you can change MD5 keys. You can do this by configuring multiple MD5 keys, each with a unique key ID, and setting the date and time to switch to the new key. Each unique MD5 key has a unique ID. The ID is used by the receiver of the OSPF packet to determine which key to use for authentication. The key identifier, which is required for MD5 authentication, specifies the identifier associated with the MD5 key.

The start time specifies when to start using the MD5 key. This is optional. The start-time option enables you to configure a smooth transition mechanism for multiple keys. The start time is relevant for transmission but not for receiving OSPF packets.

See the following sections:

- Example: Configuring IPsec Authentication for an OSPFv2 Interface on page 460
- Example: Configuring a Transition of MD5 Keys on page 461
- Example: Configuring MD5 Authentication on page 461

Example: Configuring IPsec Authentication for an OSPFv2 Interface

Configure IPsec authentication for OSPFv2 interface so-0/2/0.0. Include the name of the manual SA sa1 that you configure at the [edit security ipsec] hierarchy level.

```
[edit protocols ospf]
area 0.0.0.0 {
    interface so-0/2/0.0 {
        ipsec-sa sa-1;
    }
}
```

Example: Configuring a Transition of MD5 Keys

Configure new keys to take effect at 12:01 AM on the first day of the next three months:

```
[edit protocols ospf area 0.0.0.0 interface fe-0/0/1]
authentication {
    md5 1 {
        key $2001HaL;
    }
}
authentication {
    md5 2 {
        key NeWpsswdFEB {
            start-time 2006-02-01.00:01;
        }
    }
}
authentication {
    md5 3 {
        key NeWpsswdMAR {
            start-time 2006-03-01.00:01;
        }
    }
}
authentication {
    md5 4 {
        key NeWpsswdAPR {
            start-time 2006-04-01.00:01;
        }
    }
}
```

Set the same passwords and transition dates and times on all the routers in the area so that OSPF adjacencies remain active.

Example: Configuring MD5 Authentication

Configure MD5 authentication for OSPF:

```
[edit protocols ospf]
area 0.0.0.0 {
    interface fxp0.0 {
        disable;
    }
    interface t1-0/2/1.0 {
        authentication {
            md5 3 key "$9$6gBqCtOW87YgJEcyKW8Vb" start-time 2002-11-19.10:00;
                # SECRET-DATA
            md5 2 key "$9$DJHkP5T3/A0Uj6A0Irl"; # SECRET-DATA
        }
    }
    reference-bandwidth 4g;
```

```
    traceoptions {
        file ospf size 5m world-readable;
        flag error;
    }
}
```

Configuring Authentication for OSPFv3

OSPF version 3 (OSPFv3) provides a method for protecting and securing the OSPF traffic through the router. OSPFv3 uses the IP authentication header (AH) and the IP Encapsulating Security Payload (ESP) to authenticate routing information.

Use ESP with NULL encryption to provide authentication to the OSPFv3 protocol headers only. Use AH to provide authentication to the OSPFv3 protocol headers, portions of the IPv6 header, and portions of the extension headers. Use ESP with non-NULL encryption for full confidentiality.

OSPFv3 authentication uses static keyed IP Security (IPsec) security associations (SAs) similar to BGP IPsec. Tunnel mode SAs and dynamic IPsec SAs using Internet Key Exchange (IKE) authentication are not supported. Dynamic keyed IPsec SAs run on the Routing Engine and do not require a services PIC.

To apply authentication, include the ipsec-sa statement for a specific OSPFv3 interface:

```
interface interface-name ipsec-sa name;
```

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for these statements.

You specify the IPsec authentication name by including the *name* option. You configure the actual IPsec authentication separately.

For more information on IPsec, see the *JUNOS System Basics Configuration Guide* and the *JUNOS Services Interfaces Configuration Guide*.

Limiting the Number of Prefixes Exported to OSPF

By default, there is no limit to the number of prefixes that can be exported into OSPF. To limit the number of prefixes, include the prefix-export-limit statement:

```
prefix-export-limit number;
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

The number can be a value from 0 through 4,294,967,295.

Configuring the Designated Router Priority for OSPF

A routing device advertises its priority to become a designated router in its hello packets. On all multiaccess networks, the Hello protocol uses the advertised priorities to elect a designated router for the network. This routing device is responsible for

sending network link advertisements, which describe all the routing devices attached to the network. These advertisements are flooded throughout a single area.

At least one routing device on each logical IP network or subnet must be eligible to be the designated router for OSPFv2. At least one routing device on each logical link must be eligible to be the designated router for OSPFv3.

A routing device's priority for becoming the designated router is indicated by an arbitrary number from 0 through 255, with a higher value indicating a greater likelihood of becoming the designated router. By default, routing devices have a priority value of 128. A value of 1 means that the routing device has the least chance of becoming a designated router. A value of 0 marks the routing device as ineligible to become the designated router.

To modify the routing device's priority value, include the priority statement:

```
priority number;
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

Summarizing Ranges of Routes in OSPF Link-State Advertisements

Area border routers send summary link advertisements to describe the routes to other areas. To minimize the number of these advertisements that are flooded, you can configure the routing device to coalesce, or summarize, a range of IP addresses and send reachability information about these addresses in a single link-state advertisement.

To summarize a range of IP addresses, include the area-range statement. To summarize multiple ranges, include multiple area-range statements.

```
area area-id {
    area-range network/mask-length <restrict > <exact> <override-metric metric>;
}
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

All routes that match the specified area range are filtered at the area boundary, and the summary is advertised in their place. If you specify the restrict option, the routes are filtered but no summary is advertised. If you specify the exact option, summarization of a route is advertised only when an exact match is made with the configured summary range. To override the metric for the IP address range and configure a specific metric value, include the override-metric option. If you specify the override-metric option, the dynamically computed metric for the IP address range is overridden by the specified value.

Configuring the Metric Value for OSPF Interfaces

All OSPF interfaces have a cost, which is a routing metric that is used in the link-state calculation. Routes with lower total path metrics are preferred over those with higher path metrics.

When several equal-cost routes to a destination exist, traffic is distributed equally among them.

The cost of a route is described by a single dimensionless metric that is determined using the following formula:

```
cost = reference-bandwidth / bandwidth
```

You can modify the reference-bandwidth value. The bandwidth value refers to the actual bandwidth of the physical interface.

You can override the default behavior of using the reference bandwidth to calculate the metric cost of a route by configuring a specific metric value for any OSPF interface.

To modify the reference bandwidth, include the **reference-bandwidth** statement:

```
reference-bandwidth reference-bandwidth;
```

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for these statements.

The default value of the reference bandwidth is 100 Mbps (which you specify as 100,000,000), which gives a metric of 1 for any interface with a physical bandwidth that is 100 Mbps or greater. For *reference-bandwidth*, you can configure a value from 9600 through 1,000,000,000,000 bits.

For example, if you set the reference bandwidth to 1 Gbps (that is, *reference-bandwidth* is set to 1,000,000,000), a 100-Mbps interface has a default metric of 10.

By default, the loopback interface (**lo0**) metric is 0. No bandwidth is associated with the loopback interface.

When you specify a metric for a specific OSPF interface, that value is used to determine the cost of routes advertised from that interface. To specify a metric for routes advertised from an interface, include the **metric** statement:

```
area area-id {
    interface interface-name {
        metric metric;
    }
}
```

For *metric*, you can specify a value from 1 through 65,535.

Dynamically Adjusting OSPF Interface Metrics Based on Bandwidth

You can specify a set of bandwidth threshold values and associated metric values for an OSPF interface or for a topology on an OSPF interface. When the bandwidth of an interface changes, the JUNOS Software automatically sets the interface metric to the value associated with the appropriate bandwidth threshold value. The JUNOS Software uses the smallest configured bandwidth threshold value that is equal to or higher than the actual interface bandwidth to determine the metric value. If the interface bandwidth is higher than any of the configured bandwidth threshold values, the metric value configured for the interface is used instead of any of the bandwidth-based metric values configured. The ability to recalculate the metric for an interface when its bandwidth changes is especially useful for aggregate interfaces.

To configure bandwidth-based metrics for an OSPF interface or for a topology on an OSPF interface, include the bandwidth-based-metrics statement:

```
bandwidth-based-metrics {
    bandwidth value;
    metric number;
}
```

For the bandwidth *value* option, specify a number, in bits per second, from 9600 through 1,000,000,000,000.

For the metric *number* option, specify a value from 1 through 65,535 to associate with each bandwidth value you configure.

NOTE: You must also configure a metric for the interface when you enable bandwidth-based metrics. For more information, see "Configuring the Metric Value for OSPF Interfaces" on page 464.

Configuring Preference Values for OSPF Routes

Route preferences are used to select which route is installed in the forwarding table when several protocols calculate routes to the same destination. The route with the lowest preference value is selected. For more information about route preferences, see "Route Preferences Overview" on page 6.

By default, internal OSPF routes have a preference value of 10, and external OSPF routes have a value of 150. To change the preference values, include the preference statement (for internal routes) or the external-preference statement (for external routes):

```
external-preference preference;
preference preference;
```

For a list of hierarchy levels at which you can include these statements, see the statement summary section for these statements.

The preference can be a value from 0 through 4,294,967,295 ($2^{32} - 1$).

Configuring OSPF Timers

OSPF routing devices constantly track the status of their neighbors, sending and receiving hello packets that indicate whether the neighbor still is functioning, and sending and receiving link-state advertisement and acknowledgment packets. OSPF sends packets and expects to receive packets at specified intervals.

You can perform the following tasks when modifying the OSPF timers:

- Modifying the Hello Interval on page 466
- Controlling the LSA Retransmission Interval on page 467
- Modifying the Router Dead Interval on page 467
- Specifying the Transit Delay on page 467

Modifying the Hello Interval

Routing devices send hello packets at a fixed interval on all interfaces, including virtual links, to establish and maintain neighbor relationships. This interval, which must be the same on all routing devices on a shared network, is advertised in the hello interval field in the hello packet. By default, the routing device sends hello packets every 10 seconds.

To modify how often the routing device sends hello packets out of an interface, include the hello-interval statement:

```
hello-interval seconds;
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

On nonbroadcast networks, the routing device sends hello packets every 120 seconds until active neighbors are detected by default. This interval is long enough to minimize the bandwidth required on slow WAN links. To modify this interval, include the poll-interval statement:

```
poll-interval seconds;
```

NOTE: The poll-interval statement is valid for OSPFv2 only.

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

Once the routing device detects an active neighbor, the hello packet interval changes from the time specified in the poll-interval statement to the time specified in the hello-interval statement.

Controlling the LSA Retransmission Interval

When a routing device sends link-state advertisements to its neighbors, the routing device expects to receive an acknowledgment packet from the neighbor within a certain amount of time. If the routing device does not receive an acknowledgment, it retransmits the advertisement.

NOTE: You must configure LSA retransmit intervals to be equal or greater than 3 seconds to avoid triggering a retransmit trap because the JUNOS Software delays LSA acknowledgments by up to 2 seconds.

By default, the routing device waits 5 seconds for an acknowledgment before retransmitting the link-state advertisement. To modify this interval, include the retransmit-interval statement:

```
retransmit-interval seconds;
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

Modifying the Router Dead Interval

If a routing device does not receive a hello packet from a neighbor within a fixed amount of time, the routing device modifies its topological database to indicate that the neighbor is nonoperational. The time that the routing device waits is called the *router dead interval*. By default, this interval is 40 seconds (four times the default hello interval).

To modify the router dead interval, include the dead-interval statement. This interval must be the same for all routing devices on a shared network.

```
dead-interval seconds;
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

Specifying the Transit Delay

Before a link-state update packet is propagated out of an interface, the routing device must increase the age of the packet. If you have a very slow link (for example, one with an average propagation delay of multiple seconds), the age of the packet must be increased by a similar amount. Doing this ensures that you do not receive a packet back that is younger than the original copy.

The default transit delay is 1 second. You should never have to modify the default value. However, if you need to specify the approximate transit delay to use to age update packets, include the transit-delay statement:

```
transit-delay seconds;
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

Configuring OSPF Refresh and Flooding Reduction in Stable Topologies

The OSPF standard requires that every link-state advertisement (LSA) be refreshed every 30 minutes. The Juniper implementation refreshes LSAs every 50 mintues. By default, any LSA that is not refreshed expires after 60 minutes. This requirement can result in traffic overhead that makes it difficult to scale OSPF networks. You can override the default behavior by specifying that the DoNotAge bit be set in self-originated LSAs when they are initially sent by the router. Any LSA with the DoNotAge bit set is reflooded only when a change occurs in the LSA. This feature thus reduces protocol traffic overhead while permitting any changed LSAs to be flooded immediately. Routers enabled for flood reduction continue to send hello packets to their neighbors and to age self-originated LSAs in their databases.

The Juniper implementation of OSPF refresh and flooding reduction is based on RFC 4136, *OSPF Refresh and Flooding Reduction in Stable Topologies*. However, the Juniper implementation does not include the forced-flooding interval defined in the RFC. Not implementing the forced-flooding interval ensures that LSAs with he DoNotAge big set are reflooded only when a change occurs.

This feature is supported for the following:

- OSPFv2 and OSPFv3 interfaces
- OSPFv3 realms
- OSPFv2 and OSPFv3 virtual links
- OSPFv2 sham links
- OSPFv2 peer interfaces
- All routing instances supported by OSPF
- Logical systems

To configure flooding reduction for an OSPF interface, include the **flood-reduction** statement at the [edit protocols (ospf | ospf3) area *area-id* interface *interface-id* hierarchy level.

NOTE: If you configure flooding reduction for an interface configured as a demand circuit, the LSAs are not initially flooded, and LSAs are sent only when their content has changed. Hello packets and LSAs are sent and received on a demand-circuit interface only when a change occurs in the network topology.

In the following example, the OSPF interface **so-0/0/1.0** is configured for flooding reduction. As a result, all the LSAs generated by the routes that traverse the specified interface have the DoNotAge bit set when they are initially flooded, and LSAs are refreshed only when a change occurs.

```
[edit]
protocols ospf {
```

```
    area 0.0.0.0 {
      interface so-0/0/1.0 {
        flood-reduction;
      }
      interface lo0.0
      interface so-0/0/0.0;
    }
  }
```

Related Topics ■ flood-reduction

Configuring BFD for OSPF

The Bidirectional Forwarding Detection (BFD) protocol is a simple hello mechanism that detects failures in a network. BFD works with a wide variety of network environments and topologies. Hello packets are sent at a specified, regular interval. A neighbor failure is detected when the routing device stops receiving a reply after a specified interval. The BFD failure detection timers have shorter time limits than the OSPF failure detection mechanisms, providing faster detection. These timers are also adaptive. For example, the timer can adapt to a higher value if an adjacency fails, or a neighbor can negotiate a higher value than the one configured.

NOTE: BFD is supported for OSPFv3 in JUNOS Release 9.3 and later.

To enable failure detection, include the bfd-liveness-detection statement:

```
bfd-liveness-detection {
  detection-time {
    threshold milliseconds;
  }
  full-neighbors only;
  minimum-interval milliseconds;
  minimum-receive-interval milliseconds;
  no-adaptation;
  transmit-interval {
    threshold milliseconds;
    minimum-interval milliseconds;
  }
  multiplier number;
  version (1 | automatic);
}
```

For a list of hierarchy levels at which you can include these statements, see the reference page for the bfd-liveness-detection statement.

To specify the threshold for the adaptation of the detection time, include the threshold statement:

```
detection-time {
  threshold milliseconds;
}
```

When the BFD protocol session detection time adapts to a value equal to or greater than the threshold, a single trap and a single system message are sent.

To specify the minimum transmit and receive intervals for failure detection, include the minimum-interval statement:

```
minimum-interval milliseconds;
```

This value represents the minimum interval at which the local routing device transmits hello packets as well as the minimum interval at which the routing device expects to receive a reply from a neighbor with which it has established a BFD session. You can configure a number in the range from 1 through 255,000 milliseconds. You can also specify the minimum transmit and receive intervals separately.

NOTE: Specifying an interval less than 300 ms can cause undesired BFD flapping.

To specify only the minimum receive interval for failure detection, include the minimum-receive-interval statement:

```
minimum-receive-interval milliseconds;
```

This value represents the minimum interval at which the local routing device expects to receive a hello packet from a neighbor with which it has established a BFD session. You can configure a number in the range from 1 through 255,000 milliseconds.

To specify the number of hello packets not received by a neighbor that causes the originating interface to be declared down, include the multiplier statement:

```
multiplier number;
```

The default is 3, and you can configure a value in the range from 1 through 255.

To specify only the minimum transmit interval for failure detection, include the transmit-interval minimum-interval statement:

```
transmit-interval {
    minimum-interval milliseconds;
}
```

This value represents the minimum interval at which the local routing device transmits hello packets to the neighbor with which it has established a BFD session. You can configure a value in the range from 1 through 255,000 milliseconds.

To specify the threshold for detecting the adaptation of the transmit interval, include the threshold statement:

```
transmit-interval {
    threshold milliseconds;
}
```

The threshold value must be greater than the transmit interval.

You can trace BFD operations by including the traceoptions statement at the [edit protocols bfd] hierarchy level. For more information, see “Tracing BFD Protocol Traffic” on page 78.

In JUNOS Release 9.0 and later, you can configure BFD sessions not to adapt to changing network conditions. To disable BFD adaptation, include the no-adaptation statement:

```
no-adaptation;
```

NOTE: We recommend that you not disable BFD adaptation unless it is preferable not to have BFD adaptation enabled in your network.

In JUNOS Release 9.5 and later, you can configure the BFD protocol to establish BFD sessions only for OSPF neighbors in the full state. The default behavior is to establish BFD sessions for all OSPF neighbors. Include the full-neighbors-only statement:

```
full-neighbors-only;
```

To specify the BFD version used for detection, include the version statement:

```
version (1 | automatic);
```

The default is to have the version detected automatically.

Overview of BFD Authentication for OSPF

BFD enables rapid detection of communication failures between adjacent systems. By default, authentication for BFD sessions is disabled. However, when you run BFD over Network Layer protocols, the risk of service attacks can be significant. We strongly recommend using authentication if you are running BFD over multiple hops or through insecure tunnels. Beginning with JUNOS Release 9.6, the JUNOS Software supports authentication for BFD sessions running over OSPFv2. BFD authentication is not supported on MPLS OAM sessions. BFD authentication is only supported in the domestic image and is not available in the export image.

You authenticate BFD sessions by specifying an authentication algorithm and keychain, and then associating that configuration information with a security authentication keychain using the keychain name.

The following sections describe the supported authentication algorithms, security keychains, and level of authentication that can be configured:

- BFD Authentication Algorithms on page 472
- Security Authentication Keychains on page 472
- Strict Versus Loose Authentication on page 473

BFD Authentication Algorithms

The JUNOS Software supports the following algorithms for BFD authentication:

- simple-password—Plain-text password. One to 16 bytes of plain text are used to authenticate the BFD session. One or more passwords may be configured. This method is the least secure and should be used only when BFD sessions are not subject to packet interception.
- keyed-md5—Keyed Message Digest 5 hash algorithm for sessions with transmit and receive intervals greater than 100 ms. To authenticate the BFD session, keyed MD5 uses one or more secret keys (generated by the algorithm) and a sequence number that is updated periodically. With this method, packets are accepted at the receiving end of the session if one of the keys matches and the sequence number is greater than or equal to the last sequence number received. Although more secure than a simple password, this method is vulnerable to replay attacks. Increasing the rate at which the sequence number is updated can reduce this risk.
- meticulous-keyed-md5—Meticulous keyed Message Digest 5 hash algorithm. This method works in the same manner as keyed MD5, but the sequence number is updated with every packet. Although more secure than keyed MD5 and simple passwords, this method may take additional time to authenticate the session.
- keyed-sha-1—Keyed Secure Hash Algorithm I for sessions with transmit and receive intervals greater than 100 ms. To authenticate the BFD session, keyed SHA uses one or more secret keys (generated by the algorithm) and a sequence number that is updated periodically. The key is not carried within the packets. With this method, packets are accepted at the receiving end of the session if one of the keys matches and the sequence number is greater than the last sequence number received.
- meticulous-keyed-sha-1—Meticulous keyed Secure Hash Algorithm I. This method works in the same manner as keyed SHA, but the sequence number is updated with every packet. Although more secure than keyed SHA and simple passwords, this method may take additional time to authenticate the session.

NOTE: Nonstop active routing (NSR) is not supported with meticulous-keyed-md5 and meticulous-keyed-sha-1 authentication algorithms. BFD sessions using these algortihms may go down after a switchover.

Security Authentication Keychains

The security authentication keychain defines the authentication attributes used for authentication key updates. When the security authentication keychain is configured and associated with a protocol through the keychain name, authentication key updates can occur without interrupting routing and signaling protocols.

The authentication keychain contains one or more keychains. Each keychain contains one or more keys. Each key holds the secret data and the time at which the key becomes valid. The algorithm and keychain must be configured on both ends of the

BFD session, and they must match. Any mismatch in configuration prevents the BFD session from being created.

BFD allows multiple clients per session, and each client can have its own keychain and algorithm defined. To avoid confusion, we recommend specifying only one security authentication keychain.

Strict Versus Loose Authentication

By default, strict authentication is enabled and authentication is checked at both ends of each BFD session. Optionally, to smooth migration from nonauthenticated sessions to authenticated sessions, you can configure *loose checking*. When loose checking is configured, packets are accepted without authentication being checked at each end of the session. This feature is intended for transitional periods only.

Related Topics

- Configuring BFD Authentication for OSPF on page 473
- bfd-liveness-detection statement
- authentication-key-chains statement in the *JUNOS System Basics Configuration Guide*
- show bfd session command in the *JUNOS Routing Protocols and Policies Command Reference*
- Configuring BFD for OSPF on page 469

Configuring BFD Authentication for OSPF

Beginning with JUNOS Release 9.6, you can configure authentication for BFD sessions running over OSPFv2. Routing instances are also supported. Only three steps are needed to configure authentication on a BFD session:

1. Specify the BFD authentication algorithm for the OSPFv2 protocol.
2. Associate the authentication keychain with the OSPFv2 protocol.
3. Configure the related security authentication keychain.

The following sections provide instructions for configuring and viewing BFD authentication on OSPF:

- Configuring BFD Authentication Parameters on page 473
- Viewing Authentication Information for BFD Sessions on page 475

Configuring BFD Authentication Parameters

To configure BFD authentication:

1. Specify the algorithm (keyed-md5, keyed-sha-1, meticulous-keyed-md5, meticulous-keyed-sha-1, or simple-password) to use for BFD authentication on an OSPF route or routing instance.

   [edit]

```
user@host# set protocols ospf interface if2-ospf bfd-liveness-detection
    authentication algorithm keyed-sha-1
```

NOTE: Nonstop active routing (NSR) is not supported with meticulous-keyed-md5 and meticulous-keyed-sha-1 authentication algorithms. BFD sessions using these algortihms may go down after a switchover.

2. Specify the keychain to be used to associate BFD sessions on the specified OSPF route or routing instance with the unique security authentication keychain attributes. This should match the keychain name configured at the [edit security authentication key-chains] hierarchy level.

```
[edit]
user@host# set protocols ospf interface if2-ospf bfd-liveness-detection
    authentication keychain bfd-ospf
```

NOTE: The algorithm and keychain must be configured on both ends of the BFD session, and they must match. Any mismatch in configuration prevents the BFD session from being created.

3. Specify the unique security authentication information for BFD sessions:
   - The matching *key-chain-name* as specified in step 2.
   - At least one *key*, a unique integer between 0 and 63. Creating multiple keys allows multiple clients to use the BFD session.
   - The *secret-data* used to allow access to the session.
   - The time at which the authentication key becomes active, *yyyy-mm-dd.hh:mm:ss*.

```
[edit security]
user@host# authentication-key-chains key-chain bfd-ospf key 53 secret
    $9$ggaJDmPQ6/tJgF/AtREVsyPsnCtUHm start-time 2009-06-14.10:00:00
```

4. (Optional) Specify loose authentication checking if you are transitioning from nonauthenticated sessions to authenticated sessions.

```
[edit]
user@host> set protocols ospf interface if2-ospf bfd-liveness-detection
    authentication loose-check
```

5. (Optional) View your configuration using the show bfd session detail or show bfd session extensive command.
6. Repeat these steps to configure the other end of the BFD session.

NOTE: BFD authentication is only supported in the domestic image and is not available in the export image.

Viewing Authentication Information for BFD Sessions

You can view the existing BFD authentication configuration using the show bfd session detail and show bfd session extensive commands.

The following example shows BFD authentication configured for the if2-ospf BGP group. It specifies the keyed SHA-1 authentication algorithm and a keychain name of bfd-ospf. The authentication keychain is configured with two keys. Key 1 contains the secret data "9ggaJDmPQ6/tJgF/AtREVsyPsnCtUHm" and a start time of June 1, 2009, at 9:46:02 AM PST. Key 2 contains the secret data "9a5jiKW9I.reP38ny.TszF2/9" and a start time of June 1, 2009, at 3:29:20 PM PST.

```
[edit protocols ospf]
interface if2–ospf {
   bfd-liveness-detection {
      authentication {
         algorithm keyed-sha-1;
         key-chain bfd-ospf;
      }
   }
}
[edit security]
authentication key-chains {
   key-chain bfd-ospf {
      key 1 {
         secret "$9$ggaJDmPQ6/tJgF/AtREVsyPsnCtUHm";
         start-time "2009-6-1.09:46:02 -0700";
      }
      key 2 {
         secret "$9$a5jiKW9I.reP38ny.TszF2/9";
         start-time "2009-6-1.15:29:20 -0700";
      }
   }
}
```

If you commit these updates to your configuration, you would see output similar to the following. In the output for the show bfd sessions detail command, Authenticate is displayed to indicate that BFD authentication is configured. For more information about the configuration, use the show bfd sessions extensive command. The output for this command provides the keychain name, the authentication algorithm and mode for each client in the session, and the overall BFD authentication configuration status, keychain name, and authentication algorithm and mode.

show bfd sessions detail

```
user@host# show bfd session detail

                                                Detect   Transmit
Address                 State     Interface     Time     Interval  Multiplier
10.9.1.33               Up        so-7/1/0.0     0.600     0.200      3
  Client OSPF, TX interval 0.200, RX interval 0.200, multiplier 3, Authenticate
```

```
  Session up time 3d 00:34
  Local diagnostic None, remote diagnostic None
  Remote state Up, version 1
  Replicated

1 sessions, 1 clients
Cumulative transmit rate 10.0 pps, cumulative receive rate 10.0 pps
```

show bfd sessions extensive

```
user@host# show bfd session extensive
                                                  Detect   Transmit
Address                 State     Interface       Time     Interval  Multiplier
10.9.1.33            Up        so-7/1/0.0          0.600     0.200      3
  Client OSPF, TX interval 0.200, RX interval 0.200, multiplier 3, Authenticate

       keychain bfd-ospf, algo keyed-md5, mode loose

  Session up time 3d 00:34
  Local diagnostic None, remote diagnostic None
  Remote state Up, version 1
  Replicated
  Min async interval 0.200, min slow interval 1.000
  Adaptive async tx interval 0.200, rx interval 0.200
  Local min tx interval 0.200, min rx interval 0.200, multiplier 3
  Remote min tx interval 0.100, min rx interval 0.100, multiplier 3
  Threshold transmission interval 0.000, Threshold for detection time 0.000
  Local discriminator 11, remote discriminator 80
  Echo mode disabled/inactive
 Authentication enabled/active, keychain bfd-ospf, algo keyed-sha-1, mode strict

1 sessions, 1 clients
Cumulative transmit rate 10.0 pps, cumulative receive rate 10.0 pps
```

Related Topics

- Overview of BFD Authentication for OSPF on page 471
- bfd-liveness-detection statement
- authentication-key-chains statement in the *JUNOS System Basics Configuration Guide*
- show bfd session command in the *JUNOS Routing Protocols and Policies Command Reference*
- Configuring BFD for OSPF on page 469

Configuring Synchronization Between LDP and IGPs

LDP is a protocol for distributing labels in non-traffic-engineered applications. Labels are distributed along the best path determined by the IGP. If synchronization between LDP and the IGP is not maintained, the label-switch path (LSP) goes down. When LDP is not fully operational on a given link (a session is not established and labels are not exchanged), the IGP advertises the link with the maximum cost metric. The link is not preferred but remains in the network topology.

LDP synchronization is supported only on active point-to-point interfaces and LAN interfaces configured as point-to-point under the IGP. LDP synchronization is not supported during graceful restart.

To advertise the maximum cost metric until LDP is operational for synchronization, include the ldp-synchronization statement:

```
ldp-synchronization {
    disable;
    hold-time seconds;
}
```

To disable synchronization, include the disable statement. To configure the time period to advertise the maximum cost metric for a link that is not fully operational, include the hold-time statement.

NOTE: If you do not configure the hold-time statement, the hold-time value defaults to infinity.

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

Configuring Graceful Restart for OSPF and OSPFv3

OSPF supports two types of graceful restart: planned and unplanned. During a planned restart, the restarting routing device informs the neighbors before restarting. The neighbors act as if the routing device is still within the network topology, and continue forwarding traffic to the restarting routing device. A grace period is set to specify the time period for which the neighbors should consider the restarting routing device as part of the topology. During an unplanned restart, the routing device restarts without warning.

NOTE: On a broadcast link with a single neighbor, when the neighbor initiates an OSPFv3 graceful restart operation, the restart might be terminated at the point when the local routing device assumes the role of a helper. A change in the LSA is considered a topology change, which terminates the neighbor's restart operation.

Graceful restart is disabled by default. You can globally enable graceful restart for all routing protocols at the [edit routing-options] hierarchy level.

To configure graceful restart parameters specifically for OSPF, include the graceful-restart statement:

```
graceful-restart {
    disable;
    helper-disable;
    notify-duration seconds;
    restart-duration seconds;
}
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

To disable graceful restart, specify the disable statement. To configure a time period for complete reacquisition of OSPF neighbors, specify the restart-duration statement. To configure a time period for sending out purged grace LSAs over all interfaces, specify the notify-duration statement. Helper mode is enabled by default. To disable the graceful restart helper capability, specify the helper-disable statement.

The grace period interval for OSPF graceful restart is determined as equal to or smaller than the sum of the notify-duration time interval and the restart-duration time interval. The grace period is the number of seconds that the routing device's neighbors continue to advertise the routing device as fully adjacent, regardless of the connection state between the routing device and its neighbors.

Configuring SPF Options for OSPF

You can configure the following shortest-path-first (SPF) options:

- The delay in the time between the detection of a topology change and when the SPF algorithm actually runs.
- The maximum number of times that the SPF algorithm can run in succession before the hold-down timer begins.
- The time to hold down, or wait, before running another SPF calculation after the SPF algorithm has run in succession the configured maximum number of times.

To configure SPF options, include the spf-options statement:

```
spf-options {
    delay milliseconds;
    holddown milliseconds;
    rapid-runs number;
}
```

To configure the SPF delay, include the delay statement when specifying the spf-options statement:

```
delay milliseconds;
```

By default, the SPF algorithm runs 200 milliseconds after the detection of a topology change. The range that you can configure is from 50 through 8000 milliseconds.

To configure the maximum number of times that the SPF algorithm can run in succession, include the rapid-runs statement when specifying the spf-options statement:

```
rapid-runs number;
```

The default number of SPF calculations that can occur in succession is 3. The range that you can configure is from 1 through 5. Each SPF algorithm is run after the configured SPF delay. When the maximum number of SPF calculations occurs, the hold-down timer begins. Any subsequent SPF calculation is not run until the hold-down timer expires.

To configure the SPF hold-down timer, include the holddown statement when specifying the spf-options statement:

holddown *milliseconds*;

The default is 5000 milliseconds, and the range that you can configure is from 2000 through 20,000 milliseconds. Use the hold-down timer to hold down, or wait, before running any subsequent SPF calculations after the SPF algorithm runs for the configured maximum number of times. If the network stabilizes during the holddown period and the SPF algorithm does not need to run again, the system reverts to the configured values for the delay and rapid-runs statements.

Advertising Interface Addresses Without Running OSPF

By default, OSPF must be configured on an interface for direct interface addresses to be advertised as interior routes. To advertise the direct interface addresses without actually running OSPF on that interface, include the passive statement:

```
area area-id {
    interface interface-name {
        passive;
    }
}
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

Point-to-point interfaces differ from multipoint in that only one OSPF adjacency is possible. (A LAN, for instance, can have multiple addresses and can run OSPF on each subnet simultaneously.) As such, when you configure a numbered point-to-point interface to OSPF by name, multiple OSPF interfaces are created. One, which is unnumbered, is the interface on which the protocol is run. An additional OSPF interface is created for each address configured on the interface, if any, which is automatically marked as passive.

For OSPFv3, one OSPF-specific interface must be created per interface name configured under OSPFv3. OSPFv3 does not allow interfaces to be configured by IP address.

Enabling OSPF on an interface (by including the interface statement), disabling it (by including the disable statement), and not actually having OSPF run on an interface (by including the passive statement) are mutually exclusive states.

You can also configure interfaces in OSPF passive traffic engineering mode. For more information, see “Configuring OSPF Passive Traffic Engineering Mode” on page 479 and the *JUNOS MPLS Applications Configuration Guide*.

Configuring OSPF Passive Traffic Engineering Mode

Ordinarily, interior routing protocols such as OSPF are not run on links between ASs. However, for inter-AS traffic engineering to function properly, information about the inter-AS link—in particular, the address on the remote interface—must be made available inside the AS. This information is not normally included either in the EBGP reachability messages or in the OSPF routing advertisements.

To flood this link address information within the AS and make it available for traffic engineering calculations, you must configure OSPF passive mode for traffic engineering on each inter-AS interface. You must also supply the remote address for OSPF to distribute and include in the traffic engineering database. OSPF TE mode allows MPLS label-switched paths (LSPs) to dynamically discover OSPF AS boundary routers and to allow routers to establish a traffic engineering LSP across multiple ASs.

To configure OSPF passive mode for traffic engineering on an inter-AS interface, include the traffic-engineering statement at the [edit protocols ospf area *area-id* interface *interface-name* passive] hierarchy level:

```
[edit protocols ospf area area-id interface interface-name passive]
traffic-engineering {
   remote-node-id address /* IP address at far end of inter-AS link *
}
```

For more information, see the *JUNOS MPLS Applications Configuration Guide*.

Advertising Label-Switched Paths into OSPF

You can advertise label-switched paths (LSPs) into OSPFv2 as point-to-point links so that all participating routers can take the LSP into account when performing SPF calculations. The advertisement contains a local address (the from address of the LSP), a remote address (the to address of the LSP), and a metric with the following precedence:

1. Use the LSP metric defined under OSPFv2.
2. Use the LSP metric configured for the label-switched path under MPLS.
3. If you do not configure any of the above, use the default OSPFv2 metric of 1.

To advertise LSPs, include the label-switched-path statement, with a specified name and metric:

```
label-switched-path name metric metric;
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

NOTE: If you want an LSP that is announced into OSPFv2 to be used in SPF calculations, there must be a reverse link (that is, a link from the tail end of the LSP to the head end). You can accomplish this by configuring an LSP in the reverse direction and also announcing it in OSPFv2.

For more information about advertising label-switched paths, see the *JUNOS MPLS Applications Configuration Guide*.

Configuring OSPF to Make Routing Devices Appear Overloaded

If the time elapsed after the OSPF instance is enabled is less than the specified timeout, overload mode is set.

You can configure the local routing device so that it appears to be overloaded. You might do this when you want the routing device to participate in OSPF routing, but do not want it to be used for transit traffic. (Traffic to directly attached interfaces continues to transit the routing device.)

You configure or disable overload mode in OSPF with or without a timeout. Without a timeout, overload mode is set until it is explicitly deleted from the configuration. With a timeout, overload mode is set if the time elapsed since the OSPF instance started is less than the specified timeout.

A timer is started for the difference between the timeout and the time elapsed since the instance started. When the timer expires, overload mode is cleared. In overload mode, the router LSA is originated with all the transit router links (except stub) set to a metric of 0xFFFF. The stub router links are advertised with the actual cost of the interfaces corresponding to the stub. This causes the transit traffic to avoid the overloaded routing device and take paths around the routing device. However, the overloaded routing device's own links are still accessible.

To mark the routing device as overloaded, include the overload statement:

```
overload;
```

To specify the number of seconds at which overload is reset, include the timeout option when specifying the overload statement:

```
overload timeout seconds;
```

The time can be a value from 60 through 1800 seconds.

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for these statements.

Enabling OSPF Traffic Engineering Support

When traffic engineering is enabled on the routing device, you can enable OSPF traffic engineering support, which allows OSPF to generate LSAs that carry traffic engineering parameters. These parameters are used to create the traffic engineering database, which is used by Constrained Shortest Path First (CSPF) to compute MPLS LSPs.

NOTE: Whenever possible, use OSPF IGP shortcuts instead of traffic engineering shortcuts.

By default, traffic engineering support is disabled. To enable it, include the traffic-engineering statement:

```
traffic-engineering {
    advertise-unnumbered-interfaces;
    multicast-rpf-routes;
    no-topology;
    shortcuts {
        ignore-lsp-metrics;
        lsp-metric-into-summary;
    }
}
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

To disable the dissemination of the link-state topology information, include the no-topology statement. To use LSPs as next hops, specify the shortcuts statement.

When traffic engineering is enabled for OSPF, the SPF algorithm takes into account the various LSPs configured under MPLS. These routes are installed into the primary routing table, inet.0. To advertise the LSP metric for a prefix in a summary LSA, specify the lsp-metric-into-summary statement. To ignore RSVP LSP metrics in OSPF traffic engineering shortcut calculations, specify the ignore-lsp-metrics statement.

You can configure OSPF to install routes with regular IP next hops (no LSPs as next hops) into the inet.2 routing table for a reverse-path-forwarding (RPF) check. The inet.2 routing table consists of unicast routes used for multicast RPF lookup. RPF is an antispoofing mechanism used to check if the packet is coming in on an interface that is also sending data back to the packet source. To install routes for multicast RPF checks into the inet.2 routing table, include the multicast-rpf-routes statement.

NOTE: You must enable OSPF traffic engineering shortcuts to use the multicast-rpf-routes statement. You must not allow LSP advertisement into OSPF when configuring the multicast-rpf-routes statement.

In some scenarios, you might want to advertise the link-local identifier in the link-local TE link-state advertisement packets. To advertise unnumbered interfaces in a traffic-engineering environment, include the advertise-unnumbered-interfaces statement.

NOTE: The advertise-unnumbered-interfaces statement has no effect on your configuration if RSVP can signal unnumbered interfaces, as defined in RFC 3477, *Signalling Unnumbered Links in Resource Reservation Protocol - Traffic Engineering (RSVP-TE)*. You do not need to configure this statement in this situation.

By default, the JUNOS Software prefers IS-IS routes in the traffic engineering database over other IGP routes even if the routes of another IGP are configured with a lower, that is, more preferred, preference value. The traffic engineering database assigns a credibility value to each IGP and prefers the routes of the IGP with the highest

credibility value. In JUNOS Release 9.4 and later, you can configure OSPF to take protocol preference into account to determine the traffic engineering database credibility value. When protocol preference is used to determine the credibility value, IS-IS routes are not automatically preferred by the traffic engineering database, depending on your configuration. For example, OSPF routes have a default preference value of 10, while IS-IS Level 1 routes have a default preference value of 15. When protocol preference is enabled, the credibility value is determined by deducting the protocol preference value from a base value of 512. Using default protocol preference values, OSPF has a credibility value of 502, while IS-IS has a credibility value of 497. Because the traffic engineering database prefers IGP routes with the highest credibility value, OSPF routes are now preferred.

This feature is also supported for IS-IS. For more information see, "Configuring IS-IS Traffic Engineering Attributes" on page 351.

To configure OSPF to take protocol preference into account to determine the traffic engineering database credibility value, include the credibility-protocol-preference statement:

```
[edit protocols ospf]
traffic-engineering;
credibility-protocol-preference;
```

NOTE: The credibility protocol-preference statement is supported only for OSPFv2.

For more information about configuring LSPs and MPLS, see the *JUNOS MPLS Applications Configuration Guide*.

Example: Enabling OSPF Traffic Engineering Support

Enable OSPF traffic engineering support by configuring a virtual link on the local router. This router must be an area border router that is physically connected to the backbone.

```
[edit protocols]
ospf {
    traffic-engineering {
        shortcuts {
            lsp-metric-into-summary;
        }
    }
}
[edit protocols]
mpls {
    traffic-engineering bgp-igp;
    label-switched-path xxxx {
        to yy.yy.yy.yy;
    }
}
```

Configuring the OSPF Metric Value Used for Traffic Engineering

When traffic engineering is enabled on the router, you can configure an OSPF metric that is used exclusively for traffic engineering. The traffic engineering metric is used for information injected into the traffic engineering database. Its value does not affect normal OSPF forwarding.

To modify the default value, include the te-metric statement:

```
te-metric metric;
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

Applying Policies to OSPF Routes

All routing protocols store the routes that they learn in the routing table. The routing table uses this collected route information to determine the active routes to destinations. The routing table then installs the active routes into its forwarding table and also exports them back into the routing protocols. It is these exported routes that the protocols advertise.

For each protocol, you control which routes the protocol stores in the routing table and which routes the routing table exports into the protocol by defining a *routing policy* for that protocol. For information about defining a routing policy, see the *JUNOS Policy Framework Configuration Guide*.

By default, if a routing device has multiple OSPF areas, learned routes from other areas are automatically installed into area 0 of the routing table.

To apply routing policies that affect how the routing table exports routes into OSPF, include the export statement:

```
export [ policy-names ];
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

OSPF import policy allows users to define policy to prevent adding OSPF routes to the routing table. This filtering happens when OSPF installs the route in the routing table. You can filter the routes, but not LSA flooding. The import policy can filter on any attribute of the OSPF route.

To filter OSPF routes from being added to the routing table, include the import statement:

```
import [ policy-names ];
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

Configuring Import and Export Policies for Network Summaries

By default, OSPF uses network-summary link-state advertisements (LSAs) to transmit route information across area boundaries. Each area border router (ABR) floods network-summary LSAs to other routing devices in the same area. In JUNOS Release 9.1 and later, you can configure export and import policies for OSPFv2 and OSPFv3 that enable you to control how network-summary LSAs, which contain information about interarea OSPF prefixes, are distributed and generated. For OSPFv3, the LSA is referred to as the interarea prefix LSA and performs the same function as a network-summary LSA performs for OSPFv2. An ABR originates an interarea prefix LSA for each IPv6 prefix that must be advertised into an area.

The export policy enables you to specify which summary LSAs are flooded into an area. The import policy enables you to control which routes learned from an area are used to generate summary LSAs into other areas. You define a routing policy at the [edit policy-options policy-statement *policy–name*] hierarchy level. As with all OSPF export policies, the default for network-summary LSA export policies is to reject everything. Similarly, as with all OSPF import policies, the default for network-summary LSA import policies is to accept all OSPF routes. For more information about configuring policies, see the *JUNOS Policy Framework Configuration Guide*.

To apply an export routing policy for OSPFv2 that affects which network-summary LSAs are flooded into an area, include the network-summary-export [*policy-names*] statement:

```
area area-id {
    network-summary-export [ policy-names ];
}
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

To apply an import routing policy for OSPFv2 that affects which routes learned from an area are used to generate network-summary LSAs, include the network-summary-import [*policy-names*] statement:

```
area area-id {
    network-summary-import [ policy-names ];
}
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

To apply an export routing policy for OSPFv3 that affects which interarea prefix LSAs are flooded into an area, include the inter-area-prefix-export [*policy-names*] statement:

```
area area-id {
    inter-area-prefix-export [ policy-names ];
}
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

To apply an import routing policy for OSPFv3 that affects which routes learned from an area are used to generate interarea prefix LSAs, include the inter-area-prefix-import [*policy-names*] statement:

```
area area-id {
    inter-area-prefix-import [ policy-names ];
}
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

Configuring Priority for Prefixes in Import Policy

In a network with a large number of OSPF routes, it can be useful to control the order in which routes are updated in response to a network topology change. In JUNOS Release 9.3 and later, you can specify a priority of high, medium, or low for prefixes included in an OSPF import policy. In the event of an OSPF topology change, high priority prefixes are updated in the routing table first, followed by medium and then low priority prefixes.

OSPF import policy can only be used to set priority or to filter OSPF external routes. If an OSPF import policy is applied that results in a reject terminating action for a nonexternal route, then the reject action is ignored and the route is accepted anyway. By default, such a route is now installed in the routing table with a priority of low. This behavior prevents traffic black holes, that is, silently discarded traffic, by ensuring consistent routing within the OSPF domain.

In general, OSPF routes that are not explicitly assigned a priority are treated as priority medium, except for the following:

- Summary discard routes have a default priority of low.
- Local routes that are not added to the routing table are assigned a priority of low.
- External routes that are rejected by import policy and thus are not added to the routing table are assigned a priority of low.

To specify a priority for prefixes included in an import policy, include the priority (high | medium | low) statement at the [edit policy-options policy statement *policy-statement-name* term *term-name* then] or [edit policy-options policy-statement *policy-statement-name* then] hierarchy level.

Any available match criteria applicable to OSPF routes can be used to determine the priority. Two of the most commonly used match criteria for OSPF are the route-filter and tag statements. For more information about configuring routing policy and match conditions, see the *JUNOS Policy Framework Configuration Guide*.

Example: Configuring a Route Filter Policy to Specify Priority for Prefixes Learned Through OSPF

Configure an import routing policy, ospf-import, that enables you to specify a priority for specific prefixes learned through OSPF. Routes associated with these prefixes

are installed in the routing table in the order of the prefixes' specified priority. Routes matching 200.3.0.0/16 orlonger are installed first because they have a priority of high. Routes matching 200.2.0.0/16 orlonger are installed next because they have a priority of medium. Routes matching 200.1.0.0/16 orlonger are installed last because they have a priority of low. To apply the import policy to OSPF, include the import ospf-import statement at the [edit protocols (ospf | ospf3)] hierarchy level. For a complete list of hierarchy levels at which the import statement can be configured, see the configuration statement summary for that statement.

```
policy-options {
   policy-statement ospf-import {
      term t1 {
         from {
            route-filter 200.1.0.0/16 orlonger;
         }
         then {
            priority low;
            accept;
         }
      }
      term t2 {
         from {
            route-filter 200.2.0.0/16 orlonger;
         }
         then {
            priority medium {
            accept;
         }
      }
      tern t3 {
         from {
            route-filter 200.3.0.0/16 orlonger;
         }
         then {
            priority high;
            accept;
         }
      }
   }
}
```

Configuring OSPF Routing Table Groups

To install routes learned from OSPF routing instances into routing tables in the OSPF routing table group, include the rib-group statement:

```
rib-group group-name;
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

Configuring OSPF Sham Links

You can create an intra-area link or sham link between two provider edge (PE) routers so that the VPN backbone is preferred over the back-door link. Each sham link is identified by the combination of a local endpoint address and a remote endpoint address.

To configure a sham link, include the sham-link statement:

```
sham-link {
    local address;
}
```

To configure the local endpoint address, specify the local option.

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

To configure the remote endpoint address, include the sham-link-remote statement.

```
sham-link-remote address {
    ipsec-sa name;
    demand-circuit;
    metric metric;
}
```

To configure the OSPF interface as a demand circuit, include the demand-circuit statement. To configure the remote endpoint metric value, include the metric statement. To configure IPsec authentication for the remote endpoint of a sham link, include the ipsec-sa *name* statement.

Configuring OSPF Peer Interfaces

You can configure a peer interface for OSPF routers. Generalized MPLS (GMPLS) requires traffic engineering information to be transported through a link separate from the control channel. You establish this separate link by configuring a peer interface.

To configure a peer interface, include the peer-interface statement:

```
peer-interface interface-name {
    disable;
    dead-interval seconds;
    hello-interval seconds;
    retransmit-interval seconds;
    transit-delay seconds;
}
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

To disable the peer interface, specify the disable statement. To modify the peer interface dead interval, specify the dead-interval statement. To modify how often the router sends hello packets out of the peer interface, specify the hello-interval statement. To modify how often the peer interface retransmits the link-state advertisement, specify the retransmit-interval statement. To specify the approximate transit delay to use to age update packets, include the transit-delay statement.

For more information about configuring GMPLS, see the *JUNOS MPLS Applications Configuration Guide*.

Loop-Free Alternate Routes for OSPF Overview

Support for OSPF loop-free alternate routes essentially adds IP fast-reroute capability for OSPF. The JUNOS Software precomputes loop-free backup routes for all OSPF routes. These backup routes are preinstalled in the Packet Forwarding Engine, which performs a local repair and implements the backup path when the link for a primary next hop for a particular route is no longer available. With local repair, the Packet Forwarding Engine can correct a path failure before it receives precomputed paths from the Routing Engine. Local repair reduces the amount of time needed to reroute traffic to less than 50 milliseconds. In contrast, global repair can take up to 800 milliseconds to compute a new route. Local repair enables traffic to continue to be routed using a backup path until global repair is able to calculate a new route.

A loop-free path is one that does not forward traffic back through the router to reach a given destination. That is, a neighbor whose shortest path first to the destination traverses the router is not used as backup route to that destination. To determine loop-free alternate paths for OSPF routes, the JUNOS Software runs shortest-path-first (SPF) calculations on each one-hop neighbor. You can enable support for alternate loop-free routes on any OSPF interface. Because it is common practice to enable LDP on an interface for which OSPF is already enabled, this feature also provides support for LDP label-switched paths (LSPs.)

The level of backup coverage available through OSPF routes depends on the actual network topology and is typically less than 100 percent for all destinations on any given router. You can extend backup coverage to include RSVP LSP paths.

The JUNOS Software provides two mechanisms for route redundancy for OSPF through alternate loop-free routes:

- Link protection offers per-link traffic protection. Use link protection when you assume that only a single link might become unavailable but that the neighboring node on the primary path would still be available through another interface.
- Node-link protection establishes an alternate path through a different router altogether. Use node-link protection when you assume that access to a node is lost when a link is no longer available. As a result, the JUNOS Software calculates a backup path that avoids the primary next-hop router.

When you enable link protection or node-link protection on an OSPF interface, the JUNOS Software creates an alternate path to the primary next hop for all destination routes that traverse a protected interface.

Related Topics

- Configuring Link Protection for OSPF on page 490
- Configuring Node-Link Protection for OSPF on page 491
- Excluding an OSPF Interface as a Backup for a Protected Interface on page 492
- Configuring Backup SPF Options for Protected OSPF Interfaces on page 493
- Configuring RSVP Label-Switched Paths as Backup Paths for OSPF on page 495

Configuring Link Protection for OSPF

You can configure link protection for any interface for which OSPF is enabled. When you enable link protection, the JUNOS Software creates an alternate path to the primary next hop for all destination routes that traverse a protected interface. Use link protection when you assume that only a single link might become unavailable but that the neighboring node would still be available through another interface.

Link protection is supported on the following:

- OSPFv2 and OSPFv3 interfaces
- OSPFv3 unicast realms
- OSPFv2 unicast topologies, except for multicast topologies
- All routing instances supported by OSPFv2 and OSPFv3
- Logical systems

To configure link protection for an OSPF interface:

- Include the link-protection statement at the [edit protocols (ospf | ospf3) area *area-id* interface *interface-name* hierarchy level.

BEST PRACTICE: When you configure link protection for OSPF, you must also configure a per-packet load-balancing routing policy to ensure that the routing protocol process installs all the next hops for a given route in the routing table.

In the following example, the OSPF interface so-0/0/0.0 in area 0.0.0.0 is configured for link protection. If a link for a destination route that traverses this interface becomes unavailable, the JUNOS Software creates a loop-free backup path through another interface on the neighboring node, thus avoiding the link that is no longer available.

```
[edit]
protocols {
   ospf {
      area 0.0.0.0 {
         interface so-0/0/0.0 {
            link-protection;
         }
      }
   }
}
```

Related Topics
- Loop-Free Alternate Routes for OSPF Overview on page 489
- link-protection

Configuring Node-Link Protection for OSPF

You can configure node-link protection on any interface for which OSPF is enabled. Node-link protection establishes an alternate path through a different router altogether for all destination routes that traverse a protected interface. Node-link protection assumes that the entire router, or node, has failed. The JUNOS Sotware therefore calculates a backup path that avoids the primary next-hop router.

Node-link protection is supported on the following:

- OSPFv2 and OSPFv3 interfaces
- OSPFv3 unicast realms
- OSPFv2 unicast topologies
- All routing instances supported by OSPFv2 and OSPFv3
- Logical systems

To configure node-link protection for an OSPF interface:

- Include the node-link-protection statement at the [edit protocols (ospf | ospf3) area *area-id* interface *interface-name* hierarchy level.

BEST PRACTICE: You must also configure a per-packet load-balancing routing policy to ensure that the routing protocol process installs all the next hops for a given route in the routing table.

In the following example, the OSPF interface so-0/0/0.0 in area 0.0.0.0 is configured for node-link protection. If a link for a destination route that traverses this interface becomes unavailable, the JUNOS Software creates a loop-free backup path through a different router altogether, thus avoiding the primary next-hop router.

```
[edit]
protocols {
   ospf {
      area 0.0.0.0 {
         interface so-0/0/0.0 {
            node-link-protection;
         }
      }
   }
}
```

Related Topics
- node-link-protection
- Loop-Free Alternate Routes for OSPF Overview on page 489

Excluding an OSPF Interface as a Backup for a Protected Interface

By default, all OSPF interfaces that belong to the default instance or to a specific routing instance are eligible as backup interface for interfaces configured with link-protection or node-link protection. You can specify that any OSPF interface be excluded from functioning as a backup interface to protected interfaces.

To exclude an OSPF interface as a backup interface for a protected interface:

- Include the **no-eligible-backup** statement at the [**edit protocols** (**ospf** | **ospf3**) **area** *area-id* **interface** *interface-name*] hierarchy level.

In the following example, interface **so-0/0/0.0** has been configured to prohibit backup traffic for traffic destined for a protected interface. This means that if a neighboring next-hop path or node for a protected interface fails, interface **so-0/0/0.0** cannot be used to transmit traffic to a backup path.

```
[edit]
protocols {
   ospf {
      area 0.0.0.0 {
         interface so-0/0/0.0 {
            no-eligible-backup;
         }
      }
   }
}
```

Related Topics

- no-eligible-backup
- Loop-Free Alternate Routes for OSPF Overview on page 489
- Configuring Link Protection for OSPF on page 490
- Configuring Node-Link Protection for OSPF on page 491

Configuring Backup SPF Options for Protected OSPF Interfaces

By default, if at least one OSPF interface is configured for link-protection or node-link protection, the JUNOS Software calculates backup next hops for all the topologies in an OSPF instance. You can configure the following backup shortest-path-first (SPF) options to override the default behavior:

- Disable the calculation of backup next hops for an OSPF instance or a specific topology in an instance.
- Prevent the installation of backup next hops in the routing table or the forwarding table for an OSPF instance or a specific topology in an instance.
- Limit the calculation of backup next hops to a subset of paths as defined in RFC 5286, *Basic Specification for IP Fast Reroute: Loop-Free Alternates*

You can disable the backup SPF algorithm for an OSPF instance or specific topology in an instance. Doing so prevents the calculation of backup next hops for that OSPF instance or topology.

To disable the calculation of backup next hops for an OSPF instance or topology:

- Include the **disable** statement at the [**edit protocols** (**ospf** | **ospf3**) **backup-spf-options**] or [**edit protocols ospf backup-spf-options topology** *topology-name*] hierarchy level.

In the following example, the calculation of backup next hops is disabled for the OSPF topology **voice**.

```
[edit]
protocols {
    ospf {
        topology voice {
            backup-spf-options {
                disable;
            }
        }
    }
}
```

You can configure the router to prevent the installation of backup next hops in the routing table or the forwarding table for an OSPF instance or a specific topology in an OSPF instance. The SPF algorithm continues to calculate backup next hops, but they are not installed.

To prevent the router from installing backup next hops in the routing table or the forwarding table:

- Include the **no-install** statement at the **[edit protocols (ospf | ospf3) backup-spf-options]** or the **[edit protocols ospf topology *topology-name*]** hierarchy level.

In the following example, backup next hops for the OSPF topology **voice** are not installed in the routing table or forwarding table. Any calculated backup next hops for other OSPF instances or topologies continued to be installed.

```
[edit]
protocols {
    ospf {
        topology voice {
            backup-spf-options {
                no-install;
            }
        }
    }
}
```

You can limit the calculation of backup next hops to *downstream paths*, as defined in RFC 5286, *Basic Specification for IP Fast Reroute: Loop-Free Alternates*. You can specify for the JUNOS Software to use only downstream paths as backup next hops for protected interfaces for an OSPF instance or a specific topology in an OSPF instance. In a downstream path, the distance from the backup neighbor to the destination must be smaller than the distance from the calculating router to the destination. Using only downstream paths as loop-free alternate paths for protected interfaces ensures that these paths not result in microloops. However, you might experience less than optimal backup coverage for your network.

To limit the calculation of backup next hops to downstream paths:

- Include the **downstream-paths-only** statement at the **[edit protocols (ospf | ospf3) backup-spf-options]** or **[edit protocols ospf backup-spf-options topology *topology-name*]** hierarchy level.

In the following example, only downstream paths are calculated as backup next hops for the topology **voice**.

```
[edit]
protocols {
    ospf {
        topology voice {
            backup-spf-options {
                downstream-paths-only;
            }
        }
    }
}
```

Related Topics

- Loop-Free Alternate Routes for OSPF Overview on page 489
- backup-spf-options

- Configuring Link Protection for OSPF on page 490
- Configuring Node-Link Protection for OSPF on page 491

Configuring RSVP Label-Switched Paths as Backup Paths for OSPF

When configuring an OSPF interface for link protection or node-link protection, relying on the shortest-path-first (SPF) calculation of backup paths for one-hop neighbors might result in less than 100 percent backup coverage for a specific network topology. You can enhance coverage of OSPF and LDP label-switched-paths (LSPs) by configuring RSVP LSPs as backup paths.

When configuring an LSP, you must specify the IP address of the egress router.

NOTE: RSVP LSPs can be used as backup paths only for the default topology for OSPFv2 and not for a configured topology. Additionally, RSVP LSP cannot be used a backup paths for non-default instances for OSPFv2 or OSPFv3.

To configure a specific RSVP LSP as a backup path:

1. Include the backup statement at the [edit protocols mpls labeled-switched-path *lsp-name*] hierarchy level.
2. Specify the address of the egress router by including the to *ip-address* statement at the [edit protocols mpls label-switched-path hierarchy level.

In the following example, the RSVP LSP f-to-g is configured a backup LSP for protected OSPF interfaces. The egress router is configured with the IP address 192.168.1.4.

```
[edit]
protocols {
    mpls {
        label-switched-path f-to-g {
            to 192.168.1.4;
            backup;
        }
    }
}
```

Related Topics

- Loop-Free Alternate Routes for OSPF Overview on page 489
- Configuring Link Protection for OSPF on page 490
- Configuring Node-Link Protection for OSPF on page 491

Tracing OSPF Protocol Traffic

You can trace various OSPF protocol traffic to help debug OSPF protocol issues. To trace OSPF protocol traffic include the traceoptions statement at the [edit protocols ospf|ospf3] hierarchy level:

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

You can specify the following OSPF protocol-specific trace options using the flag statement:

- database-description—All database description packets, which are used in synchronizing the OSPF topological database
- error—OSPF error packets
- event—OSPF state transitions
- flooding—Link-state flooding packets
- graceful-restart—Graceful-restart events.
- hello—Hello packets, which are used to establish neighbor adjacencies and to determine whether neighbors are reachable
- ldp-synchronization—Synchronization events between OSPF and LDP
- lsa-ack—Link-state acknowledgement packets, which are used in synchronizing the OSPF topological database
- lsa-analysis—Link-state analysis packets
- lsa-request—Link-state request packets, which are used in synchronizing the OSPF topological database
- lsa-update—Link-state updates packets, which are used in synchronizing the OSPF topological database
- nsr-synchronization—Nonstop routing synchronization events
- on-demand—Trace demand circuit extensions
- packet-dump—Dump the contents of selected packet types
- packets—All OSPF packets
- spf—Shortest path first (SPF) calculations

You can optionally specify one or more of the following flag modifiers:

- detail—Detailed trace information
- receive—Packets being received
- send—Packets being transmitted

NOTE: Use the detail flag modifier with caution as this may cause the CPU to become very busy.

Global tracing options are inherited from the configuration set by the traceoptions statement at the [edit routing-options] hierarchy level. You can override the following global trace options for the OSPF protocol using the traceoptions flag statement included at the [edit protocols ospf] hierarchy level:

- all—All tracing operations
- general—All normal operations and routing table changes (a combination of the normal and route trace operations)
- normal—Normal events
- policy—Policy processing
- route—Routing information
- state—State transitions
- task—Routing protocol task processing
- timer—Routing protocol timer processing

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

Examples: Tracing OSPF Protocol Traffic

Trace only unusual or abnormal operations to the file routing-log, and trace detailed information about all OSPF packets to the file ospf-log:

```
[edit]
routing-options {
   traceoptions {
      file routing-log;
   }
}
protocols {
   ospf {
      traceoptions {
         file ospf-log size 10k files 5;
         flag lsa-ack;
         flag database-description;
         flag hello;
         flag lsa-update;
         flag lsa-request;
      }
      area 0.0.0.0 {
         interface 10.0.0.1;
      }
   }
}
```

Trace SPF calculations:

```
[edit]
```

```
protocols {
   ospf {
      traceoptions {
         file ospf-log;
         flag spf;
      }
      area 0.0.0.0 {
         interface 10.0.0.1;
      }
   }
}
```

Trace the creation, receipt, and retransmission of all link-state advertisements:

```
[edit]
protocols {
   ospf {
      traceoptions {
         file ospf-log;
         flag lsa-request;
         flag lsa-update;
         flag lsa-ack;
         area 0.0.0.0 {
            interface 10.0.0.1;
         }
      }
   }
}
```

Related Topics

- traceoptions statement
- For general information about tracing and global tracing options, see Tracing Global Routing Protocol Operations on page 129.

Chapter 19

Summary of OSPF Configuration Statements

The following sections explain each of the OSPF configuration statements, which are organized alphabetically. The term OSPF refers to both OSPF version 2 (OSPFv2) and OSPF version 3 (OSPFv3).

area

Syntax area *area-id*;

Hierarchy Level
[edit logical-systems *logical-system-name* protocols (ospf | ospf3)],
[edit logical-systems *logical-system-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast)],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols (ospf | ospf3)],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast)],
[edit protocols (ospf | ospf3)],
[edit protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast)],
[edit routing-instances *routing-instance-name* protocols (ospf | ospf3)],
[edit routing-instances *routing-instance-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast)]

Release Information
Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.
Support for the realm statement introduced in JUNOS Release 9.2.
Support for the realm statement introduced in JUNOS Release 9.2 for EX Series switches.

Description Specify the area identifier for this routing device to use when participating in OSPF routing. All routing devices in an area must use the same area identifier to establish adjacencies.

Specify multiple area statements to configure the routing device as an area border router. An area border router does not automatically summarize routes between areas; use the area-range statement to configure route summarization. By definition, an area border router must be connected to the backbone area either through a physical link or through a virtual link. To create a virtual link, include the virtual-link statement.

To specify that the routing device is directly connected to the OSPF and OSPFv3 backbone, include the area 0.0.0.0 statement.

All routing devices on the backbone must be contiguous. If they are not, use the virtual-link statement to create the appearance of connectivity to the backbone.

Options *area-id*—Area identifier. The identifier can be up to 32 bits. It is common to specify the area number as a simple integer or an IP address. Area number 0.0.0.0 is reserved for the OSPF and OSPFv3 backbone area.

Required Privilege Level
routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- virtual-link
- Configuring OSPF Areas on page 449
- Configuring Multiple Address Families for OSPFv3 on page 457

area-range

Syntax area-range *network/mask-length* <exact> <override-metric *metric*> <restrict>;

Hierarchy Level [edit logical-systems *logical-system-name* protocols (ospf | ospf3) area *area-id*],
[edit logical-systems *logical-system-name* protocols (ospf | ospf3) area *area-id* nssa],
[edit logical-systems *logical-system-name* realm (ipv4-unicast | ipv4-multicast | ipv6-multicast) area *area-id*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols (ospf | ospf3) area *area-id*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols (ospf | ospf3) area *area-id* nssa],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* realm (ipv4-unicast | ipv4-multicast | ipv6-multicast) area *area-id*],
[edit protocols (ospf | ospf3) area *area-id*],
[edit protocols (ospf | ospf3) area *area-id* nssa],
[edit protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast) area *area-id*],
[edit routing-instances *routing-instance-name* protocols (ospf | ospf3) area *area-id*],
[edit routing-instances *routing-instance-name* protocols (ospf | ospf3) area *area-id* nssa],
[edit routing-instances *routing-instance-name* realm (ipv4-unicast | ipv4-multicast | ipv6-multicast) area *area-id*]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.
Support for the realm statement introduced in JUNOS Release 9.2.
Support for the realm statement introduced in JUNOS Release 9.2 for EX Series switches.

Description (Area border routers only) For an area, summarize a range of IP addresses when sending summary link advertisements (within an area). To summarize multiple ranges, include multiple area-range statements.

For a not-so-stubby area (NSSA), summarize a range of IP addresses when sending NSSA link-state advertisements. The specified prefixes are used to aggregate external routes learned within the area when the routes are advertised to other areas. To specify multiple prefixes, include multiple area-range statements. All external routes learned within the area that do not fall into one of the prefixes are advertised individually to other areas.

Default By default, area border routers do not summarize routes being sent from one area to other areas, but rather send all routes explicitly.

Options exact—(Optional) Summarization of a route is advertised only when an exact match is made with the configured summary range.

mask-length—Number of significant bits in the network mask.

network—IP address. You can specify one or more IP addresses.

override-metric *metric*—(Optional) Override the metric for the IP address range and configure a specific metric value.

restrict—(Optional) Do not advertise the configured summary. This hides all routes that are contained within the summary, effectively creating a route filter.
Range: 1 through 16,777,215

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics ■ Summarizing Ranges of Routes in OSPF Link-State Advertisements on page 463

authentication

Syntax

```
authentication {
    md5 key-identifier {
        key key-value
        start-time YYYY-MM-DD.hh:mm
    }
    simple-password key;
}
```

Hierarchy Level

[edit logical-systems *logical-system-name* protocols ospf area *area-id* interface *interface-name*],
[edit logical-systems *logical-system-name* protocols ospf area *area-id* virtual-link],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ospf area *area-id* interface *interface-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ospf area *area-id* virtual-link],
[edit protocols ospf area *area-id* interface *interface-name*],
[edit protocols ospf area *area-id* virtual-link],
[edit routing-instances *routing-instance-name* protocols ospf area *area-id* interface *interface-name*],
[edit routing-instances *routing-instance-name* protocols ospf area *area-id* virtual-link]

Release Information Statement introduced before JUNOS Release 7.4.

Description Configure an authentication key (password). Neighboring routers use the password to verify the authenticity of packets sent from this interface.

All routers that are connected to the same IP subnet must use the same authentication scheme and password.

The remaining statements are explained separately.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics ■ Configuring Authentication for OSPFv2 on page 458

backup-spf-options

Syntax

```
backup-spf options {
    disable;
    downstream-paths-only;
    no-install;
}
```

Hierarchy Level

[edit protocols (ospf | ospf3)],
[edit logical-systems *logical-system-name* protocols (ospf | ospf3)],
[edit routing-instances *routing-instance-name* protocols (ospf | ospf3)],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols (ospf | ospf3)],
[edit protocols ospf topology (default | *name*)],
[edit logical-systems *logical-system-name* protocols ospf topology (default | *name*)],
[edit routing-instances *routing-instance-name* protocols ospf topology (default | *name*)],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ospf topology (default | *name*)];
[edit protocols ospf3 realm ipv4–unicast],
[edit logical-systems *logical-system-name* protocols ospf3 realm ipv4–unicast],
[edit routing-instances *routing-instance-name* protocols ospf3 realm ipv4–unicast],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ospf3 realm ipv4–unicast]

Release Information

Statement introduced in JUNOS Release 10.0.

Description

Configure options for running the shortest-path-first (SPF) algorithm for backup next hops for protected OSPF interfaces. Use these options to override the default behavior of having the JUNOS Software calculate backup paths for all the topologies in an OSPF instance when at least one OSPF interface is configured with link protection or node-link protection. These options also enable you to change the default behavior for a specific topology in an OSPF instance.

Options

disable—Do not calculate backup next hops for the specified OSPF instance or topology.

downstream-paths-only—Calculate and install only downstream paths as defined in RFC 5286, *Basic Specification for IP Fast Reroute: Loop-Free Alternates* for the specified OSPF instance or topology.

no-install—Do not install the backup next hops for the specified OSPF instance or topology.

Required Privilege Level

routing—To view this statement in the configuration.
routing-control-level—To add this statement to the configuration.

Related Topics

- Configuring Backup SPF Options for Protected OSPF Interfaces on page 493
- link-protection
- node-link-protection

bandwidth-based-metrics

Syntax

```
bandwidth-based-metrics {
    bandwidth value;
    metric number;
}
```

Hierarchy Level

[edit logical-systems *logical-system-name* protocols (ospf | ospf3) area *area-id* interface *interface-name*],
[edit logical-systems *logical-system-name* protocols ospf area *area-id* interface *interface-name* topology *topology-name*],
[edit logical-systems *logical-system-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast) area *area-id* interface *interface-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols (ospf | ospf3) area *area-id* interface *interface-name*],
[edit logical-systems *logical-system-name* protocols ospf area *area-id* interface *interface-name* topology *topology-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instances* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast) area *area-id* interface *interface-name*],
[edit protocols (ospf | ospf3) area *area-id* interface *interface-name*],
[edit protocols ospf area *area-id* interface *interface-name* topology *topology-name*],
[edit protocols ospf3 realm (ivp4-unicast | ipv4-multicast | ipv6-multicast) area *area-id* interface *interface-name*],
[edit routing-instances *routing-instance-name* protocols (ospf | ospf3) area *area-id* interface *interface-name*],
[edit routing-instances *routing-instance-name* protocols ospf area *area-id* interface *interface-name* topology *topology-name*],
[edit routing-instances *routing-instance-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast) area *area-id* interface *interface-name*]

Release Information

Statement introduced in JUNOS Release 9.5.
Statement introduced in JUNOS Release 9.5 for EX Series switches.

Description

Specify a set of bandwidth threshold values and associated metric values for an OSPF interface or for a topology on an OSPF interface. When the bandwidth of an interface changes, the JUNOS Software automatically sets the interface metric to the value associated with the appropriate bandwidth threshold value.

Options

bandwidth *value*—Specify the bandwidth threshold in bits per second.
Range: 9600 through 1,000,000,000,000,000

metric *number*—Specify a metric value to associate with a specific bandwidth value.
Range: 1 through 65,535

NOTE: You must also configure a static metric value for the OSPF interface or topology with the metric statement. The JUNOS Software uses this value to calculate the cost of a route from the OSPF interface or topology if the bandwidth for the interface is higher than of any bandwidth threshold values configured for bandwidth-based metrics.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- metric
- Dynamically Adjusting OSPF Interface Metrics Based on Bandwidth on page 465

bfd-liveness-detection

Syntax

```
bfd-liveness-detection {
    authentication {
        algorithm algorithm-name;
        key-chain key-chain-name;
        loose-check;
    }
    detection-time {
        threshold milliseconds;
    }
    full-neighbors-only
    minimum-interval milliseconds;
    minimum-receive-interval milliseconds;
    no-adaptation;
    transmit-interval {
        threshold milliseconds;
        minimum-interval milliseconds;
    }
    multiplier number;
    version (1 | automatic);
}
```

Hierarchy Level

[edit logical-systems *logical-system-name* protocols (ospf | ospf3) area *area-id* interface *interface-name*],
[edit logical-systems *logical-system-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast) area *area-id* interface *interface-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols (ospf | ospf3) area *area-id* interface *interface-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast) area *area-id* interface *interface-name*],
[edit protocols (ospf | ospf3) area *area-id* interface *interface-name*],
[edit protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast) area *area-id* interface *interface-name*],
[edit routing-instances *routing-instance-name* protocols (ospf | ospf3) area *area-id* interface *interface-name*],
[edit routing-instances *routing-instance-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast) area *area-id* interface *interface-name*]

Release Information

Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.
detection-time threshold and **transmit-interval threshold** options added in JUNOS Release 8.2.
Support for logical systems introduced in JUNOS Release 8.3.
no-adaptation statement introduced in JUNOS Release 9.0.
no-adaptation statement introduced in JUNOS Release 9.0 for EX Series switches.
Support for OSPFv3 introduced in JUNOS Release 9.3.
Support for OSPFv3 introduced in JUNOS Release 9.3 for EX Series switches.
full-neighbors-only statement introduced in JUNOS Release 9.5.
full-neighbors-only statement introduced in JUNOS Release 9.5 for EX Series switches.

authentication algorithm, authentication key-chain, and authentication loose-check statements introduced in JUNOS Release 9.6.

Description Configure bidirectional failure detection timers and authentication.

Options authentication algorithm *algorithm-name* —Configure the algorithm used to authenticate the specified BFD session: simple-password, keyed-md5, keyed-sha-1, meticulous-keyed-md5, or meticulous-keyed-sha-1.

authentication key-chain *key-chain-name*—Associate a security key with the specified BFD session using the name of the security keychain. The name you specify must match one of the keychains configured in the authentication-key-chains key-chain statement at the [edit security] hierarchy level.

authentication loose-check—(Optional) Configure loose authentication checking on the BFD session. Use only for transitional periods when authentication may not be configured at both ends of the BFD session.

detection-time threshold *milliseconds*—Configure a threshold. When the BFD session detection time adapts to a value equal to or greater than the threshold, a single trap and a single system log message are sent.

full-neighbors-only—Establish BFD sessions only for OSPF neighbors in the full state. The default behavior is to establish BFD sessions for all OSPF neighbors.

minimum-interval *milliseconds*—Configure the minimum intervals at which the local routing device transmits a hello packet and then expects to receive a reply from the neighbor with which it has established a BFD session.
Range: 1 through 255,000 milliseconds

minimum-receive-interval *milliseconds*—Configure only the minimum interval at which the routing device expects to receive a reply from a neighbor with which it has established a BFD session.
Range: 1 through 255,000 milliseconds

multiplier *number*—Configure the number of hello packets not received by a neighbor that causes the originating interface to be declared down.
Range: 1 through 255
Default: 3

no-adaptation—Specify that BFD sessions should not adapt to changing network conditions. We recommend that you not disable BFD adaptation unless it is preferable not to have BFD adaptation enabled in your network.

transmit-interval threshold *milliseconds*—Configure a threshold. When the BFD session transmit interval adapts to a value greater than the threshold, a single trap and a single system log message are sent. The interval threshold must be greater than the minimum transmit interval.
Range: 0 through 4,294,967,295 ($2^{32} - 1$)

transmit-interval minimum-interval *milliseconds*—Configure the minimum interval at which the routing device transmits hello packets to a neighbor with which it has established a BFD session.
Range: 1 through 255,000

version—Specify the BFD version to detect.
Range: 1 (BFD version 1) or automatic (autodetect version)
Default: automatic

The remaining statements are explained separately.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring BFD for OSPF on page 469
- Configuring BFD Authentication for OSPF on page 473

dead-interval

Syntax dead-interval *seconds*;

Hierarchy Level [edit logical-systems *logical-system-name* protocols ospf area *area-id* peer-interface *interface-name*],
[edit logical-systems *logical-system-name* protocols (ospf | ospf3) area *area-id* interface *interface-name*],
[edit logical-systems *logical-system-name* protocols (ospf | ospf3) area *area-id* virtual-link],
[edit logical-systems *logical-system-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast) area *area-id* interface *interface-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols (ospf | ospf3) area *area-id* interface *interface-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols (ospf | ospf3) area *area-id* virtual-link],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast) area *area-id* interface *interface-name*],
[edit protocols ospf area *area-id* peer-interface *interface-name*],
[edit protocols (ospf | ospf3) area *area-id* interface *interface-name*],
[edit protocols (ospf | ospf3) area *area-id* virtual-link],
[edit protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast) area *area-id* interface *interface-name*],
[edit routing-instances *routing-instance-name* protocols (ospf | ospf3) area *area-id* interface *interface-name*],
[edit routing-instances *routing-instance-name* protocols (ospf | ospf3) area *area-id* virtual-link],
[edit routing-instances *routing-instance-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast) area *area-id* interface *interface-name*]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.
Support for the realm statement introduced in JUNOS Release 9.2.
Support for the realm statement introduced in JUNOS Release 9.2 for EX Series switches.

Description Specify how long OSPF waits before declaring that a neighboring routing device is unavailable. This is an interval during which the routing device receives no hello packets from the neighbor.

Options *seconds*—Interval to wait.
Range: 1 through 65,535 seconds
Default: 40 seconds (four times the hello interval)

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- hello-interval
- Modifying the Router Dead Interval on page 467

default-lsa

Syntax

```
default-lsa {
    default-metric metric;
    metric-type type;
    type-7;
}
```

Hierarchy Level

[edit logical-systems *logical-system-name* protocols (ospf | ospf3) area *area-id* nssa],
[edit logical-systems *logical-system-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast) area *area-id* nssa],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols (ospf | ospf3) area *area-id* nssa],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast) area *area-id* nssa],
[edit protocols (ospf | ospf3) area *area-id* nssa],
[edit protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast) area *area-id* nssa],
[edit routing-instances *routing-instance-name* protocols (ospf | ospf3) area *area-id* nssa],
[edit routing-instances *routing-instance-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast) area *area-id* nssa]

Release Information

Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.
Support for the realm statement introduced in JUNOS Release 9.2.
Support for the realm statement introduced in JUNOS Release 9.2 for EX Series switches.

Description

On area border routers only, for an NSSA, inject a default LSA with a specified metric value into the area. The default route matches any destination that is not explicitly reachable from within the area.

The remaining statements are explained separately.

Required Privilege Level

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

Related Topics

- nssa
- stub
- Configuring OSPF Not-So-Stubby Areas on page 451

default-metric

Syntax default-metric *metric*;

Hierarchy Level
[edit logical-systems *logical-system-name* protocols (ospf | ospf3) area *area-id* nssa default-lsa],
[edit logical-systems *logical-system-name* protocols (ospf | ospf3) area *area-id* stub],
[edit logical-systems *logical-system-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast) area *area-id* nssa default-lsa],
[edit logical-systems *logical-system-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast) area *area-id* stub],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols (ospf | ospf3) area *area-id* nssa default-lsa],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols (ospf | ospf3) area *area-id* stub],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast) area *area-id* nssa default-lsa],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast) area *area-id* stub],
[edit protocols (ospf | ospf3) area *area-id* nssa default-lsa],
[edit protocols (ospf | ospf3) area *area-id* stub],
[edit protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast) area *area-id* nssa default-lsa],
[edit protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast) area *area-id* stub],
[edit routing-instances *routing-instance-name* protocols (ospf | ospf3) area *area-id* nssa default-lsa],
[edit routing-instances *routing-instance-name* protocols (ospf | ospf3) area *area-id* stub],
[edit routing-instances *routing-instance-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast) area *area-id* nssa default-lsa],
[edit routing-instances *routing-instance-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast) area *area-id* stub]

Release Information
Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.
Support for the realm statement introduced in JUNOS Release 9.2.
Support for the realm statement introduced in JUNOS Release 9.2 for EX Series switches.

Description On area border routers only, for a stub area, inject a default route with a specified metric value into the area. The default route matches any destination that is not explicitly reachable from within the area.

Options *metric*—Metric value.
Range: 1 through 16,777,215

Required Privilege Level
routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- nssa
- stub
- Configuring OSPF Stub Areas on page 450

demand-circuit

Syntax demand-circuit;

Hierarchy Level
[edit logical-systems *logical-system-name* protocols (ospf | ospf3) area *area-id* interface *interface-name*],
[edit logical-systems *logical-system-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast) area *area-id* interface *interface-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ospf area *area-id* sham-link-remote],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols (ospf | ospf3) area *area-id* interface *interface-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast) area *area-id* interface *interface-name*],
[edit protocols (ospf | ospf3) area *area-id* interface *interface-name*],
[edit protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast) area *area-id* interface *interface-name*],
[edit routing-instances *routing-instance-name* protocols (ospf | ospf3) area *area-id* interface *interface-name*],
[edit routing-instances *routing-instance-name* protocols ospf area *area-id* sham-link-remote],
[edit routing-instances *routing-instance-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast) area *area-id* interface *interface-name*]

Release Information Statement introduced before JUNOS Release 7.4.
Support for the realm statement introduced in JUNOS Release 9.2.

Description Configure an interface as a demand circuit.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring an OSPF Demand Circuit Interface on page 456
- Configuring OSPF Sham Links on page 488

disable

See the following sections:

- disable (LDP Synchronization) on page 516
- disable (OSPF) on page 517

disable (LDP Synchronization)

Syntax disable;

Hierarchy Level [edit logical-systems *logical-system-name* protocols (ospf | ospf3) area *area-id* interface *interface-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols (ospf | ospf3) area *area-id* interface *interface-name*],
[edit protocols (ospf | ospf3) area *area-id* interface *interface-name*],
[edit routing-instances *routing-instance-name* protocols (ospf | ospf3) area *area-id* interface *interface-name*]

Release Information Statement introduced in JUNOS Release 7.5.

Description Disable LDP for OSPF.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring Synchronization Between LDP and IGPs on page 476

disable (OSPF)

Syntax disable;

Hierarchy Level
[edit logical-systems *logical-system-name* protocols (ospf | ospf3)],
[edit logical-systems *logical-system-name* protocols (ospf | ospf3) area *area-id* interface *interface-name*],
[edit logical-systems *logical-system-name* protocols (ospf | ospf3) virtual-link],
[edit logical-systems *logical-system-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast)],
[edit logical-systems *logical-system-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast) area *area-id* interface *interface-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols (ospf | ospf3)],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols (ospf | ospf3) area *area-id* interface *interface-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols (ospf | ospf3) virtual-link],
[edit logical-systems *logical-system-name* routing-instances *routing-instances* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast)],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast) area *area-id* interface *interface-name*],
[edit protocols (ospf | ospf3)],
[edit protocols (ospf | ospf3) area *area-id* interface *interface-name*],
[edit protocols (ospf | ospf3) virtual-link],
[edit protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast)],
[edit protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast) area *area-id* interface *interface-name*],
[edit routing-instances *routing-instance-name* protocols (ospf | ospf3)],
[edit routing-instances *routing-instance-name* protocols (ospf | ospf3) area *area-id* interface *interface-name*],
[edit routing-instances *routing-instance-name* protocols (ospf | ospf3) virtual-link],
[edit routing-instances *routing-instance-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast)],
[edit routing-instances *routing-instance-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast) area *area-id* interface *interface-name*]

Release Information
Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.
Support for the realm statement introduced in JUNOS Release 9.2.
Support for the realm statement introduced in JUNOS Release 9.2. for EX Series switches.

Description Disable OSPF, an OSPF interface, or an OSPF virtual link.

Default The configured object is enabled (operational) unless explicitly disabled.

Required Privilege Level
routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Minimum OSPF Configuration on page 449

domain-id

Syntax domain-id *domain-id*;

Hierarchy Level [edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols (ospf | ospf3)],
[edit routing-instances *routing-instance-name* protocols (ospf | ospf3)]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Specify a domain ID for a route. The domain ID identifies the OSPF domain from which the route originated.

Options *domain-id*—You can specify either an IP address or an IP address and a local identifier using the following format: *ip-address:local-identifier*. If you do not specify a local identifier with the IP address, the identifier is assumed to have a value of 0.
Default: If the router ID is not configured in the routing instance, the router ID is derived from an interface address belonging to the routing instance.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring OSPF Domain IDs for VPNs on page 266

domain-vpn-tag

Syntax domain-vpn-tag *number*;

Hierarchy Level [edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols (ospf | ospf3)],
[edit routing-instances *routing-instance-name* protocols (ospf | ospf3)]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Set a virtual private network (VPN) tag for OSPFv2 external routes generated by the provider edge (PE) router.

Options *number*—VPN tag.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring OSPF Domain IDs for VPNs on page 266

export

Syntax export [*policy–names*];

Hierarchy Level
[edit logical-systems *logical-system-name* protocols (ospf | ospf3)],
[edit logical-systems *logical-system-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast)],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols (ospf | ospf3)],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast)],
[edit protocols (ospf | ospf3)],
[edit protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast)],
[edit routing-instances *routing-instance-name* protocols (ospf | ospf3)],
[edit routing-instances *routing-instance-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast)]

Release Information
Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.
Support for the realm statement introduced in JUNOS Release 9.2.
Support for the realm statement introduced in JUNOS Release 9.2 for EX Series switches.

Description Apply one or more policies to routes being exported from the routing table into OSPF.

Options *policy-names*—Name of one or more policies.

Required Privilege Level
routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Applying Policies to OSPF Routes on page 484
- *JUNOS Policy Framework Configuration Guide*

external-preference

Syntax external-preference *preference*;

Hierarchy Level
[edit logical-systems *logical-system-name* protocols (ospf | ospf3)],
[edit logical-systems *logical-system-name* protocols ospf3 realm (ip4-unicast | ipv4-multicast | ipv6-multicast)],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols (ospf | ospf3)],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast)],
[edit protocols (ospf | ospf3)],
[edit protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast)],
[edit routing-instances *routing-instance-name* protocols (ospf | ospf3)],
[edit routing-instances *routing-instance-name* protocols ospf3 realm (ipv4-unicast } ipv4-multicast | ipv6-multicast)]

Release Information
Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.
Support for the realm statement introduced in JUNOS Release 9.2.
Support for the realm statement introduced in JUNOS Release 9.2 for EX Series switches.

Description Set the route preference for OSPF external routes.

Options *preference*—Preference value.
Range: 0 through 4,294,967,295 ($2^{32} - 1$)
Default: 150

Required Privilege Level
routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- preference
- Configuring Preference Values for OSPF Routes on page 465

flood-reduction

Syntax flood-reduction;

Hierarchy Level [edit protocols (ospf | ospf3) area *area-id* interface *interface-name*],
[edit logical-systems *logical-system-name* protocols (ospf | ospf3) area *area-id* interface *interface-name*],
[edit routing-instances *routing-instance-name* protocols (ospf | ospf3) area *area-id* interface *interface-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols (ospf | ospf3) area *area-id* interfaces *interface-name*],
[edit protocols ospf3 realm (ipv4-multicast | ipv4-unicast | ipv6-multicast) area *area-id* interfaces *interface-name*],
[edit logical-systems *logical-system-name* protocols ospf3 realm (ipv4-multicast | ipv4-unicast | ipv6-multicast) area *area-id* interfaces *interface-name*],
[edit routing-instances *routing-instance-name* protocols ospf3 realm (ipv4-multicast | ipv4-unicast | ipv6-multicast) area *area-id* interfaces *interface-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ospf3 realm (ipv4-multicast | ipv4-unicast | ipv6-multicast) area *area-id* interfaces *interface-name*],
[edit protocols (ospf | ospf3) area *area-id* virtual-link neighbor-id *router-id* transit-area *area-id*],
[edit logical-systems *logical-system-name* protocols (ospf | ospf3) area *area-id* virtual-link neighbor-id *router-id* transit–area *transit-area* area-id],
[edit routing-instances *routing-instance-name* protocols (ospf | ospf3) area *area-id* virtual-link neighbor-id *router-id* transit-area *area-id*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols (ospf | ospf3) area *area-id* virtual-link neighbor-id *router-id* transit-area *area-id*],
[edit routing-instances *routing-instance-name* protocols ospf area *area-id* sham-link-remote *address*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ospf area area-id sham-link-remote *address*],
[edit protocols ospf area *area-id* peer-interface *interface-name*],
[edit logical-systems *logical-system-name* protocols ospf area *area-id* peer-interface *interface-name*]

Release Information Statement introduced in JUNOS Release 9.6.

Description Specify to send self-generated link-state advertisements (LSAs) with the DoNot Age bit set. As a result, self-originated LSAs are not reflooded every 30 minutes, as required by OSPF by default. An LSA is refreshed only when the content of the LSA changes, which reduces OSPF traffic overhead in stable topologies.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring OSPF Refresh and Flooding Reduction in Stable Topologies on page 468

graceful-restart

Syntax

```
graceful-restart {
    disable;
    helper-disable;
    notify-duration seconds;
    restart-duration seconds;
}
```

Hierarchy Level [edit logical-systems *logical-system-name* protocols (ospf | ospf3)],
[edit protocols (ospf | ospf3)]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Configure graceful restart for OSPF.

Options

disable—Disable graceful restart for OSPF.

helper-disable—Disable graceful restart helper capability. Helper mode is enabled by default.

notify-duration *seconds*—Estimated time to send out purged grace LSAs over all the interfaces.
Range: 1 through 3600 seconds
Default: 30 seconds

restart-duration *seconds*—Estimated time to reacquire a full OSPF neighbor from each area.
Range: 1 through 3600 seconds
Default: 180 seconds

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics

- Configuring Graceful Restart for OSPF and OSPFv3 on page 477
- *JUNOS High Availability Configuration Guide*

hello-interval

Syntax hello-interval *seconds*;

Hierarchy Level
[edit logical-systems *logical-system-name* protocols ospf area *area-id* peer-interface *interface-name*],
[edit logical-systems *logical-system-name* protocols (ospf | ospf3) area *area-id* interface *interface-name*],
[edit logical-systems *logical-system-name* protocols (ospf | ospf3) area *area-id* virtual-link],
[edit logical-systems *logical-system-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast) area *area-id* interface *interface-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols (ospf | ospf3) area *area-id* interface *interface-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols (ospf | ospf3) area *area-id* virtual-link],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast) area *area-id* interface *interface-name*],
[edit protocols ospf area *area-id* peer-interface *interface-name*],
[edit protocols (ospf | ospf3) area *area-id* interface *interface-name*],
[edit protocols (ospf | ospf3) area *area-id* virtual-link],
[edit protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast) area *area-id* interface *interface-name*],
[edit routing-instances *routing-instance-name* protocols (ospf | ospf3) area *area-id* interface *interface-name*],
[edit routing-instances *routing-instance-name* protocols (ospf | ospf3) area *area-id* virtual-link],
[edit routing-instances *routing-instance-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast) area *area-id* interface *interface-name*]

Release Information
Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.
Support for the realm statement introduced in JUNOS Release 9.2.
Support for the realm statement introduced in JUNOS Release 9.2 for EX Series switches.

Description Specify how often the routing device sends hello packets out the interface. The hello interval must be the same for all routing devices on a shared logical IP network.

Options
seconds—Time between hello packets, in seconds.
Range: 1 through 255 seconds
Default: 10 seconds; 120 seconds (nonbroadcast networks)

Required Privilege Level
routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- dead-interval
- Modifying the Hello Interval on page 466

hold-time

Syntax hold-time *seconds*;

Hierarchy Level [edit logical-systems *logical-system-name* protocols ospf area *area-id* interface *interface-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ospf area *area-id* interface *interface-name*],
[edit protocols ospf area *area-id* interface *interface-name*],
[edit routing-instances *routing-instance-name* protocols ospf area *area-id* interface *interface-name*]

Release Information Statement introduced in JUNOS Release 7.5.

Description Configure the time period to advertise the maximum cost metric for a link that is not fully operational.

Options *seconds*—Hold-time value.
Range: 1 through 65,535 seconds
Default: Infinity

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring Synchronization Between LDP and IGPs on page 476

ignore-lsp-metrics

Syntax ignore-lsp-metrics;

Hierarchy Level [edit logical-systems *logical-system-name* protocols ospf traffic-engineering shortcuts],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ospf traffic-engineering shortcuts],
[edit protocols ospf traffic-engineering shortcuts],
[edit routing-instances *routing-instance-name* protocols ospf traffic-engineering shortcuts]

Release Information Statement introduced in JUNOS Release 7.5.
Statement introduced in JUNOS Release 9.0 for EX Series switches.
Support for (OSPFv3) introduced in JUNOS Release 9.4.
Support for (OSPFv3) introduced in JUNOS Release 9.4 for EX Series switches.

Description Ignore RSVP LSP metrics in OSPF traffic engineering shortcut calculations.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Enabling OSPF Traffic Engineering Support on page 481

import

Syntax import [*policy–names*];

Hierarchy Level
[edit logical-systems *logical-system-name* protocols (ospf | ospf3)],
[edit logical-systems *logical-system-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast)],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols (ospf | ospf3)],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast)],
[edit protocols (ospf | ospf3)],
[edit protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast)],
[edit routing-instances *routing-instance-name* protocols (ospf | ospf3)],
[edit routing-instances *routing-instance-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast)]

Release Information
Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.
Support for the realm statement introduced in JUNOS Release 9.2.
Support for the realm statement introduced in JUNOS Release 9.2 for EX Series switches.

Description Filter OSPF routes from being added to the routing table.

Options *policy-names*—Name of one or more policies.

Required Privilege Level
routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Applying Policies to OSPF Routes on page 484
- *JUNOS Policy Framework Configuration Guide*

inter-area-prefix-export

Syntax inter-area-prefix-export [*policy-names*];

Hierarchy Level [edit logical-systems *logical-system-name* protocols ospf3 area *area-id*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ospf3 area *area-id*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ospf3 realm (ip4-unicast | ipv4-multicast | ipv6-multicast) area *area-id*
[edit protocols ospf3 area *area-id*],
[edit protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast) area *area-id*],
[edit routing-instances *routing-instance-name* protocols ospf3 area *area-id*],
[edit routing-instances *routing-instance-name* protocols ospf3 realm (ipv4-unicast | ipv4-muticast | ipv6-multicast) area *area-id*]

Release Information Statement introduced in JUNOS Release 9.1.
Statement introduced in JUNOS Release 9.1 for EX Series switches.
Support for the realm statement introduced in JUNOS Release 9.2.
Support for the realm statement introduced in JUNOS Release 9.2 for EX Series switches.

Description Apply an export policy for OSPFv3 to specify which interarea prefix link-state advertisements (LSAs) are flooded into an area.

Options *policy-name*—Name of a policy configured at the [edit policy-options policy-statement *policy-name* term *term-name*] hierarchy level.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- inter-area-prefix-import
- Configuring Import and Export Policies for Network Summaries on page 485
- *JUNOS Policy Framework Configuration Guide*

inter-area-prefix-import

Syntax inter-area-prefix-import [*policy-names*];

Hierarchy Level
[edit logical-systems *logical-system-name* protocols ospf3 area *area-id*],
[edit logical-systems *logical-system-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast) area *area-id*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ospf3 area *area-id*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast) area *area-id*],
[edit protocols ospf3 area *area-id*],
[edit protocols ospf3 realm (ip4-unicast | ipv4-multicast | ipv6-multicast)], area *area-id*],
[edit routing-instances *routing-instance-name* protocols ospf3 area *area-id*],
[edit routing-instances *routing-instance-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast) area *area-id*]

Release Information
Statement introduced in JUNOS Release 9.1.
Statement introduced in JUNOS Release 9.1 for EX Series switches.
Support for the realm statement introduced in JUNOS Release 9.2.
Support for the realm statement introduced in JUNOS Release 9.2 for EX Series switches.

Description Apply an import policy for OSPFv3 to specify which routes learned from an area are used to generate interarea prefixes into other areas.

Options *policy-name*—Name of a policy configured at the [edit policy-options policy-statement *policy-name* term *term-name*] hierarchy level.

Required Privilege Level
routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- inter-area-prefix-export
- Configuring Import and Export Policies for Network Summaries on page 485
- *JUNOS Policy Framework Configuration Guide*

interface

Syntax

```
interface interface-name {
    disable;
    authentication key <key-id identifier>;
    bfd-liveness-detection {
        authentication {
            algorithm algorithm-name;
            key-chain key-chain-name;
            loose-check;
        }
        detection-time {
            threshold milliseconds;
        }
        minimum-interval milliseconds;
        minimum-receive-interval milliseconds;
        transmit-interval {
            threshold milliseconds;
            minimum-interval milliseconds;
        }
        multiplier number;
    }
    dead-interval seconds;
    demand-circuit;
    hello-interval seconds;
    ipsec-sa name;
    interface-type type;
    ldp-synchronization {
        disable;
        hold-time seconds;
    }
    metric metric;
    neighbor address <eligible>;
    passive;
    poll-interval seconds;
    priority number;
    retransmit-interval seconds;
    te-metric metric;
    topology (ipv4-multicast | name) {
        metric metric;
    }
    transit-delay seconds;
    transmit-interval seconds;
}
```

Hierarchy Level

[edit logical-systems *logical-system-name* protocols (ospf | ospf3) area *area-id*],
[edit logical-systems *logical-system-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast) area *area-id*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols (ospf | ospf3) area *area-id*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast) area *area-id*],

[edit protocols (ospf | ospf3) area *area-id*],
[edit protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast) area *area-id*],
[edit routing-instances *routing-instance-name* protocols (ospf | ospf3) area *area-id*],
[edit routing-instances *routing-instance-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast) area *area-id*]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.
Support for the topology statement introduced in JUNOS Release 9.0.
Support for the topology statement introduced in JUNOS Release 9.0 for EX Series switches.
Support for the realm statement introduced in JUNOS Release 9.2.
Support for the realm statement introduced in JUNOS Release 9.2 for EX Series switches.

Description Enable OSPF routing on a routing device interface.

You must include at least one interface statement in the configuration to enable OSPF on the routing device.

Options *interface-name*—Name of the interface. Specify the interface by IP address or interface name for OSPFv2, or only the interface name for OSPFv3. Using both the interface name and IP address of the same interface produces an invalid configuration. To configure all interfaces, you can specify all. Specifying a particular interface and all produces an invalid configuration. For details about specifying interfaces, see interface naming in the *JUNOS Network Interfaces Configuration Guide*.

NOTE: For nonbroadcast interfaces, specify the IP address of the nonbroadcast interface as *interface-name*.

The remaining statements are explained separately.

NOTE: You cannot run both OSPF and ethernet-tcc encapsulation between two Juniper Networks routing devices.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- neighbor
- Minimum OSPF Configuration on page 449
- Configuring Interface Properties for MT-OSPF on page 294
- Configuring Multiple Address Families for OSPFv3 on page 457

interface-type

Syntax interface-type (nbma | p2mp | p2p);

Hierarchy Level
[edit logical-systems *logical-system-name* protocols (ospf | ospf3) area *area-id* interface *interface-name*],
[edit logical-systems *logical-system-name* protocols ospf3 realm (ipv4-multicast | ipv4-unicast | ipv6-multicast) area *area-id* interface *interface-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols (ospf | ospf3) area *area-id* interface *interface-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ospf3 realm (ipv4-multicast | ipv4-unicast | ipv6-multicast) area *area-id* interface *interface-name*],
[edit protocols (ospf | ospf3) area *area-id* interface *interface-name*],
[edit protocols ospf3 realm (ipv4-multicast | ipv4-unicast | ipv6-multicast) area *area-id* interface *interface-name*]
[edit routing-instances *routing-instance-name* protocols (ospf | ospf3) area *area-id* interface *interface-name*],
[edit routing-instances *routing-instance-name* protocols ospf3 realm (ipv4-multicast | ipv4-unicast | ipv6-multicast) area *area-id* interface *interface-name*],

Release Information
Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.
Support for OSPFv3 for interface type p2p only introduced in JUNOS Release 9.4. You cannot configure other interface types for OSPFv3.
Support for OSPFv3 for interface type p2p only introduced in JUNOS Release 9.4 for EX Series switches.

Description Specify the type of interface.

By default, the software chooses the correct interface type based on the type of physical interface. Therefore, you should never have to set the interface type. The exception to this is for NBMA interfaces, which default to an interface type of point-to-multipoint. To have these interfaces explicitly run in NBMA mode, configure the nbma interface type, using the IP address of the local ATM interface.

In JUNOS Release 9.3 and later, a point-to-point interface can be an Ethernet interface without a subnet. For more information about configuring interfaces, see the *JUNOS Network Interfaces Configuration Guide*.

Default The software chooses the correct interface type based on the type of physical interface.

Options nbma (OSPFv2 only)—Nonbroadcast multiaccess (NBMA) interface.

p2mp (OSPFv2 only)—Point-to-multipoint interface.

p2p—Point-to-point interface.

Required Privilege Level
routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring an Interface on a Broadcast or Point-to-Point Network on page 454

ipsec-sa

Syntax
ipsec-sa *name*;

Hierarchy Level
[edit logical-systems *logical-system-name* protocols (ospf | ospf3) area *area-id* interface *interface-name*],
[edit logical-systems *logical-system-name* protocols (ospf | ospf3) area *area-id* virtual-link],
[edit logical-systems *logical-system-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast) area *area-id* interface *interface-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols (ospf | ospf3) area *area-id* interface *interface-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ospf area *area-id* sham-link-remote *address*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols (ospf | ospf3) area *area-id* virtual-link],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast) area *area-id* interface *interface-name*],
[edit protocols (ospf | ospf3) area *area-id* interface *interface-name*],
[edit protocols (ospf | ospf3) area *area-id* virtual-link],
[edit protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast) area *area-id* interface *interface-name*],
[edit routing-instances *routing-instance-name* protocols (ospf | ospf3) area *area-id* interface *interface-name*],
[edit routing-instances *routing-instance-name* protocols ospf area *area-id* sham-link-remote *address*],
[edit routing-instances *routing-instance-name* protocols (ospf | ospf3) area *area-id* virtual-link],
[edit routing-instances *routing-instance-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast) area *area-id* interface *interface-name*]

Release Information
Statement introduced before JUNOS Release 7.4.
Support for OSPFv2 authentication added in JUNOS Release 8.3.
Support for the realm statement introduced in JUNOS Release 9.2.

Description
Apply IPsec authentication to an OSPF interface or virtual link or to an OSPFv2 remote sham link.

Options
name—Name of the IPsec authentication scheme.

Required Privilege Level
routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring Authentication for OSPFv2 on page 458
- Configuring Authentication for OSPFv3 on page 462
- *JUNOS System Basics Configuration Guide*
- *JUNOS Services Interfaces Configuration Guide*
- *JUNOS Feature Guide*

label-switched-path

Syntax label-switched-path *name* metric *metric*;

Hierarchy Level [edit logical-systems *logical-system-name* protocols ospf area *area-id*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ospf area *area-id*],
[edit protocols ospf area *area-id*],
[edit routing-instances *routing-instance-name* protocols ospf area *area-id*]

Release Information Statement introduced before JUNOS Release 7.4.

Description Advertise label-switched paths into OSPF as point-to-point links.

The label-switched path is advertised in the appropriate OSPF levels as a point-to-point link and contains a local address and a remote address.

Options *name*—Name of the label-switched path.

metric—Metric value.
Range: 1 through 65,535
Default: 1

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Advertising Label-Switched Paths into OSPF on page 480

ldp-synchronization

Syntax

```
ldp-synchronization {
    disable;
    hold-time seconds;
}
```

Hierarchy Level

[edit logical-systems *logical-system-name* protocols ospf area *area-id* interface *interface-name*],
[edit logical-systems *logical-system-name* protocols ospf3 realm ipv4-unicast area *area-id* interface *interface-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ospf area *area-id* interface *interface-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ospf3 realm ipv4-unicast area *area-id* interface *interface-name*],
[edit protocols ospf area *area-id* interface *interface-name*],
[edit protocols ospf3 realm ipv4-unicast area *area-id* interface *interface-name*],
[edit routing-instances *routing-instance-name* protocols ospf area *area-id* interface *interface-name*],
[edit routing-instances *routing-instance-name* protocols ospf3 realm ipv4-unicast area *area-id* interface *interface-name*]

Release Information

Statement introduced in JUNOS Release 7.5.
Support for the realm statement introduced in JUNOS Release 9.2. Only the ipv4-unicast option is supported with this statement.

Description

Enable synchronization by advertising the maximum cost metric until LDP is operational on the link.

The remaining statements are explained separately.

Required Privilege Level

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

Related Topics

- Configuring Synchronization Between LDP and IGPs on page 476

link-protection

Syntax link-protection;

Hierarchy Level [edit protocols (ospf | ospf3) area *area-name* interface *interface-name*],
[edit logical-systems *logical-system-name* protocols (ospf | ospf3) area *area-name* interface *interface-name*],
[edit routing-instances *routing-instance-name* protocols (ospf | ospf3) area *area-name* interface *interface-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols (ospf | ospf3) area *area-name* interface *interface-name*],
[edit protocols ospf3 realm ipv4-unicast area *area-id*],
[edit logical-systems *logical-system-name* protocols ospf3 realm ipv4-unicast area *area-id*],
[edit routing-instances *routing-instance-name* protocols ospf3 realm ipv4-unicast area *area-id*],
[edit protocols ospf area *area-id* interface *interface-name* topology (default | *name*)],
[edit logical-systems *logical-system-name* protocols ospf area *area-id* interface *interface-name* topology (default | *name*)],
[edit routing-instances *routing-instance-name* protocols ospf area *area-id* interface *interface-name* topology (default | *name*)]

Release Information Statement introduced in JUNOS Release 10.0.

Description Enable link protection on the specified OSPF interface. The JUNOS Software creates a backup loop-free alternate path to the primary next hop for all destination routes that traverse the protected interface.

NOTE: This feature calculates alternate next hop paths for unicast routes only. Therefore, this statement is not supported with the OSPF IPv4 multicast topology or with the OSPFv3 IPv4 multicast and IPv6 multicast realms.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring Link Protection for OSPF on page 490
- node-link-protection

lsp-metric-into-summary

Syntax lsp-metric-into-summary;

Hierarchy Level [edit logical-systems *logical-system-name* protocols (ospf | ospf3) traffic-engineering shortcuts],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols (ospf | ospf3) traffic-engineering shortcuts],
[edit protocols (ospf | ospf3) traffic-engineering shortcuts],
[edit routing-instances *routing-instance-name* protocols (ospf | ospf3) traffic-engineering shortcuts]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.
Support for OSPFv3 (ospf3) introduced in JUNOS Release 9.4.
Support for OSPFv3 (ospf3) introduced in JUNOS Release 9.4 for EX Series switches.

Description Advertise the LSP metric in summary LSAs.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Enabling OSPF Traffic Engineering Support on page 481

md5

Syntax

```
md5 key-identifier {
    key key-values;
    start-time time;
}
```

Hierarchy Level

[edit logical-systems *logical-system-name* protocols ospf area *area-id* interface *interface-name* authentication],
[edit logical-systems *logical-system-name* protocols ospf area *area-id* virtual-link authentication],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ospf area *area-id* interface *interface-name* authentication],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ospf area *area-id* virtual-link authentication],
[edit protocols ospf area *area-id* interface *interface-name* authentication],
[edit protocols ospf area *area-id* virtual-link authentication],
[edit routing-instances *routing-instance-name* protocols ospf area *area-id* interface *interface-name* authentication],
[edit routing-instances *routing-instance-name* protocols ospf area *area-id* virtual-link authentication]

Release Information

Statement introduced before JUNOS Release 7.4.

Description

Configure an MD5 authentication key (password).

Options

key-id—MD5 key identifier.
Default: 0
Range: 0 through 255

key *key-values*—One or more MD5 key strings. The MD5 key values can be from 1 through 16 characters long. You can specify more than one key value within the list. Characters can include ASCII strings. If you include spaces, enclose all characters in quotation marks (“ ”).

start-time *time*—MD5 start time.

Required Privilege Level

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

Related Topics

- Configuring Authentication for OSPFv2 on page 458

metric

Syntax metric *metric*;

Hierarchy Level [edit logical-systems *logical-system-name* protocols (ospf | ospf3) area *area-id* interface *interface-name*],
[edit logical-systems *logical-system-name* protocols ospf area *area-id* interface *interface-name* topology (ipv4-multicast | *name*)],
[edit logical-systems *logical-system-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast) area *area-id* interface *interface-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols (ospf | ospf3) area *area-id* interface *interface-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ospf area *area-id* sham-link-remote],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ospf area *area-id* interface *interface-name* topology (ipv4-multicast | *name*)],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast) area *area-id* interface *interface-name*],
[edit protocols (ospf | ospf3) area *area-id* interface *interface-name*],
[edit protocols ospf area *area-id* interface *interface-name* topology (ipv4-multicast | *name*)],
[edit protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast) area *area-id* interface *interface-name*],
[edit routing-instances *routing-instance-name* protocols (ospf | ospf3) area *area-id* interface *interface-name*],
[edit routing-instances *routing-instance-name* protocols ospf area *area-id* sham-link-remote],
[edit routing-instances *routing-instance-name* protocols ospf area *area-id* interface *interface-name* topology (ipv4-multicast | *name*)],
[edit routing-instances *routing-instance-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast) area *area-id* interface *interface-name*]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.
Support for Multitopology Routing introduced in JUNOS Release 9.0.
Support for Multitopology Routing introduced in JUNOS Release 9.0 for EX Series switches.
Support for the realm statement introduced in JUNOS Release 9.2.
Support for the realm statement introduced in JUNOS Release 9.2 for EX Series switches.

Description Specify the cost of an OSPF interface. The cost is a routing metric that is used in the link-state calculation.

To set the cost of routes exported into OSPF, configure the appropriate routing policy.

Options *metric*—Cost of the route.

Range: 1 through 65,535

Default: By default, the cost of an OSPF route is calculated by dividing the reference-bandwidth value by the bandwidth of the physical interface. Any specific value you configure for the metric overrides the default behavior of using the reference-bandwidth value to calculate the cost of route for that interface.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- bandwidth-based-metrics
- reference-bandwidth
- Configuring the Metric Value for OSPF Interfaces on page 464
- Configuring OSPF Sham Links on page 488
- Configuring Interface Properties for MT-OSPF on page 294

metric-type

Syntax metric-type *type*;

Hierarchy Level
[edit logical-systems *logical-system-name* protocols (ospf | ospf3) area *area-id* nssa default-lsa],
[edit logical-systems *logical-system-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast)] area *area-id* nssadefault-lsa],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols (ospf | ospf3) area *area-id* nssa default-lsa],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast)] area *area-id* nssa default-lsa],
[edit protocols (ospf | ospf3) area *area-id* nssa default-lsa],
[edit protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast)] area *area-id* nssa default-lsa],
[edit routing-instances *routing-instance-name* protocols (ospf | ospf3) area *area-id* nssa default-lsa],
[edit routing-instances *routing-instance*s protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast)] area *area-id* nssa default-lsa]

Release Information
Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.
Support for the realm statement introduced in JUNOS Release 9.2.
Support for the realm statement introduced in JUNOS Release 9.2 for EX Series switches.

Description Specify the external metric type for the default LSA.

Options *type*—Metric type: 1 or 2

Required Privilege Level
routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring OSPF Not-So-Stubby Areas on page 451

neighbor

Syntax neighbor *address* <eligible>;

Hierarchy Level
[edit logical-systems *logical-system-name* protocols (ospf | ospf3) area *area-id* interface *interface-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols (ospf | ospf3) area *area-id* interface *interface-name*],
[edit protocols (ospf | ospf3) area *area-id* interface *interface-name*],
[edit routing-instances *routing-instance-name* protocols (ospf | ospf3) area *area-id* interface *interface-name*]

Release Information Statement introduced before JUNOS Release 7.4.

Description For nonbroadcast interfaces only, specify neighboring routers. On a nonbroadcast interface, you must specify neighbors explicitly because OSPF does not send broadcast packets to dynamically discover their neighbors. To specify multiple neighbors, include multiple neighbor statements.

Options *address*—IP address of a neighboring router.

eligible—(Optional) Allow the neighbor to become a designated router.
Default: If you omit this option, the neighbor is not considered eligible to become a designated router.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring an Interface on a Nonbroadcast, Multiaccess Network on page 455

network-summary-export

Syntax network-summary-export *policy-name*;

Hierarchy Level [edit logical-systems *logical-system-name* protocols ospf area *area-id*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ospf area *area-id*],
[edit protocols ospf area *area-id*],
[edit routing-instances *routing-instance-name* protocols ospf area *area-id*]

Release Information Statement introduced in JUNOS Release 9.1.

Description Apply an export policy that specifies which network-summary link-state advertisements (LSAs) are flooded into an OSPFv2 area.

Options *policy-name*—Name of a policy configured at the [edit policy-options policy-statement *policy-name* term *term-name*] hierarchy level.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring Import and Export Policies for Network Summaries on page 485
- *JUNOS Policy Framework Configuration Guide*

network-summary-import

Syntax network-summary-import *policy-name*;

Hierarchy Level [edit logical-systems *logical-system-name* protocols ospf area *area-id*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ospf area *area-id*],
[edit protocols ospf area *area-id*],
[edit routing-instances *routing-instance-name* protocols ospf area *area-id*]

Release Information Statement introduced in JUNOS Release 9.1.

Description Apply an import policy that specifies which routes learned from an OSPFv2 area are used to generate network-summary link-state advertisements to other areas.

Options *policy-name*—Name of a policy configured at the [edit policy-options policy-statement *policy-name* term *term-name*] hierarchy level.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring Import and Export Policies for Network Summaries on page 485
- *JUNOS Policy Framework Configuration Guide*

no-eligible-backup

Syntax no-eligbile-backup;

Hierarchy Level [edit protocols (ospf | ospf3) area *area-id* interface *interface-name*],
[edit logical-systems *logical-system-name* protocols (ospf | ospf3) area *area-id* interface *interface-name*],
[edit routing-instances *routing-instance-name* protocols (ospf | ospf3) area *area-id* interface *interface-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols (ospf | ospf3) area *area-id* interface *interface-name*],
[edit protocols ospf3 realm ipv4-unicast area *area-id* interface *interface-name*],
[edit logical-systems *logical-system-name* protocols ospf3 realm ipv4-unicast area *area-id* interface *interface-name*],
[edit routing-instances *routing-instance-name* protocols ospf3 realm ipv4-unicast area *area-id* interface *interface-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ospf3 realm ipv4-unicast area *area-id* interface *interface-name*],
[edit protocols ospf area *area-id* interface *interface-name* topology (default | *name*)],
[edit logical-systems *logical-system-name* protocols ospf area *area-id* interface *interface-name* topology (default | *name*)],
[edit routing-instances *routing-instance-name* protocols ospf area *area-id* interface *interface-name* topology (default | *name*)],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ospf area *area-id* interface *interface-name* topology (default | *name*)],

Release Information Statement introduced in JUNOS Release 10.0.

Description Exclude the specified interface as a backup interface for OSPF interfaces on which link protection or node-link protection is enabled.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Excluding an OSPF Interface as a Backup for a Protected Interface on page 492
- link-protection
- node-link-protection

no-nssa-abr

Syntax no-nssa-abr;

Hierarchy Level
[edit logical-systems *logical-system-name* protocols (ospf | ospf3)],
[edit logical-systems *logical-system-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast)],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols (ospf | ospf3)],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast)],
[edit protocols (ospf | ospf3)],
[edit protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast)],
[edit routing-instances *routing-instance-name* protocols (ospf | ospf3)],
[edit routing-instances *routing-instance-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast)]

Release Information
Statement introduced in JUNOS Release 7.6.
Statement introduced in JUNOS Release 9.0 for EX Series switches.
Support for the realm statement introduced in JUNOS Release 9.2.
Support for the realm statement introduced in JUNOS Release 9.2 for EX Series switches.

Description Disable exporting Type 7 link-state advertisements into not-stubby-areas (NSSAs) for an autonomous system boundary router (ASBR) or an area border router (ABR).

Required Privilege Level
routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Disabling Export of LSAs into NSSAs Attached to ASBR ABRs on page 453

no-rfc-1583

Syntax no-rfc-1583;

Hierarchy Level
[edit logical-systems *logical-system-name* protocols (ospf | ospf3)],
[edit logical-systems *logical-system-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast)],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols (ospf | ospf3)],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast)],
[edit protocols (ospf | ospf3)],
[edit protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast)],
[edit routing-instances *routing-instance-name* protocols (ospf | ospf3)],
[edit routing-instances *routing-instance-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast)]

Release Information
Statement introduced in JUNOS Release 8.5.
Statement introduced in JUNOS Release 9.0 for EX Series switches.
Support for the realm statement introduced in JUNOS Release 9.2.
Support for the realm statement introduced in JUNOS Release 9.2 for EX Series switches.

Description Disable compatibility with RFC 1583, *OSPF Version 2*. If the same external destination is advertised by AS boundary routers that belong to different OSPF areas, disabling compatibility with RFC 1583 can prevent routing loops.

Default Compatibility with RFC 1583 is enabled by default.

Required Privilege Level
routing—To view this statement in the configuration.
routing-control-level—To add this statement to the configuration.

Related Topics
- Disabling OSPFv2 Compatibility with RFC 1583 on page 453

no-summaries

See summaries

node-link-protection

Syntax node-link-protection;

Hierarchy Level [edit protocols (ospf | ospf3) protocols area *area-id* interface *interface-name*],
[edit logical-systems *logical-system-name* protocols (ospf | ospf3) area *area-id* interface *interface-name*],
[edit routing-instances *routing-instance-name* protocols (ospf | ospf3) area *area-id* interface *interface-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols (ospf | ospf3) area *area-id* interface *interface-name*],
[edit protocols ospf3 realm ipv4-unicast area *area-id* interface *interface-name*],
[edit logical-systems *logical-system-name* ospf3 realm ipv4-unicast area *area-id* interface *interface-name*],
[edit routing-instances *routing-instance-name* ospf3 realm ipv4-unicast area *area-id* interface *interface-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* ospf3 realm ipv4-unicast area *area-id* interface *interface-name*],

Release Information Statement introduced in JUNOS Release 10.0.

Description Enable node-link protection on the specified OSPF interface. The JUNOS Software creates an alternate loop-free path to the primary next hop for all destination routes that traverse a protected interface. This alternate path avoids the primary next-hop router altogether and establishes a path through a different router.

NOTE: This feature is not supported for the OSPF IPv4 multicast topology or for the OSPFv3 IPv4 multicast or IPv6 multicast topologies because node-link protection creates alternate next-hop paths only for unicast routes.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring Node-Link Protection for OSPF on page 491
- link-protection

nssa

Syntax

```
nssa {
   area-range network/mask-length <restrict> <exact> <override-metric metric>;
   default-lsa {
      default-metric metric;
      metric-type type;
      type-7;
   }
   (no-summaries | summaries);
}
```

Hierarchy Level

[edit logical-systems *logical-system-name* protocols (ospf | ospf3) area *area-id*],
[edit logical-systems *logical-system-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast)],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols (ospf | ospf3) area *area-id*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast)],
[edit protocols (ospf | ospf3) area *area-id*],
[edit protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast)],
[edit routing-instances *routing-instance-name* protocols (ospf | ospf3) area *area-id*],
[edit routing-instances *routing-instance-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast)]

Release Information

Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.
Support for the realm statement introduced in JUNOS Release 9.2.
Support for the realm statement introduced in JUNOS Release 9.2 for EX Series switches.

Description

Configure a not-so-stubby area (NSSA). An NSSA allows external routes to be flooded within the area. These routes are then leaked into other areas.

You cannot configure an area as being both a stub area and an NSSA.

The remaining statements are explained separately.

Required Privilege Level

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

Related Topics

- stub
- Configuring OSPF Not-So-Stubby Areas on page 451

ospf

Syntax ospf { ... }

Hierarchy Level [edit logical-systems *logical-system-name* protocols],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols],
[edit protocols],
[edit routing-instances *routing-instance-name* protocols]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Enable OSPF routing on the routing device.

You must include the ospf statement to enable OSPF on the routing device.

Default OSPF is disabled on the routing device.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Minimum OSPF Configuration on page 449

ospf3

Syntax ospf3 { ... }

Hierarchy Level [edit logical-systems *logical-system-name* protocols],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols],
[edit protocols],
[edit routing-instances *routing-instance-name* protocols]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Enable OSPFv3 routing on the routing device.

You must include the ospf3 statement to enable OSPFv3.

Default OSPFv3 is disabled on the routing device.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Minimum OSPF Configuration on page 449

overload

Syntax

```
overload {
    timeout seconds;
}
```

Hierarchy Level

[edit logical-systems *logical-system-name* protocols (oospf | ospf3)],
[edit logical-systems *logical-system-name* protocols ospf topology (default | ipv4-multicast | *name*)],
[edit logical-systems *logical-system-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast)],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols (ospf | ospf3)],
[edit logical systems *logical-system-name* routing-instances *routing-instance-name* protocols ospf topology (default | ipv4-multicast | *name*)],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast)],
[edit protocols (ospf | ospf3)],
[edit protocols ospf topology (default | ipv4-multicast | *name*)],
[edit protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast)],
[edit routing-instances *routing-instance-name* protocols (ospf | ospf3)]
[edit routing-instances *routing-instance-name* protocols ospf topology (default | ipv4-multicast | *name*],
[edit routing-instances *routing-instance-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast)],

Release Information

Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.
Support for Multitopology Routing introduced in JUNOS Release 9.0.
Support for Multitopology Routing introduced in JUNOS Release 9.0 for EX Series switches.
Support for the realm statement introduced in JUNOS Release 9.2.
Support for the realm statement introduced in JUNOS Release 9.2. for EX Series switches.

Description

Configure the local routing device so that it appears to be overloaded. You might do this when you want the routing device to participate in OSPF routing, but do not want it to be used for transit traffic. Note that traffic destined to immediately attached interfaces continues to reach the routing device.

Options

timeout *seconds*—(Optional) Number of seconds at which the overloading is reset. If no timeout interval is specified, the routing device remains in overload state until the overload statement is deleted or a timeout is set.
Range: 60 through 1800 seconds
Default: 0 seconds

NOTE: Multitopology Routing does not support the timeout option.

Required Privilege Level

routing—To view this statement in the configuration.

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

Related Topics

- Configuring OSPF to Make Routing Devices Appear Overloaded on page 481
- Configuring a Topology to Appear Overloaded on page 294

passive

Syntax

```
passive {
    traffic-engineering {
        remote-node-id address;
    }
}
```

Hierarchy Level

[edit logical-systems *logical-system-name* protocols (ospf | ospf3) area *area-id* interface *interface-name*],
[edit logical-systems *logical-system-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast) area *area-id* interface *interface-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols (ospf | ospf3) area *area-id* interface *interface-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast) area *area-id* interface *interface-name*],
[edit protocols (ospf | ospf3) area *area-id* interface *interface-name*],
[edit protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast) area *area-id* interface *interface-name*],
[edit routing-instances *routing-instance-name* protocols (ospf | ospf3) area *area-id* interface *interface-name*],
[edit routing-instances *routing-instance-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast) area *area-id* interface *interface-name*]

Release Information

Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.
traffic-engineering and remote-node-id *address* statements introduced in JUNOS Release 8.0.
traffic-engineering and remote-node-id *address* statements introduced in JUNOS Release 8.0 for EX Series switches.
Support for the realm statement introduced in JUNOS Release 9.2.
Support for the realm statement introduced in JUNOS Release 9.2 for EX Series switches.

Description

Advertise the direct interface addresses on an interface without actually running OSPF on that interface. A passive interface is one for which the address information is advertised as an internal route in OSPF, but on which the protocol does not run.

To configure an interface in OSPF passive traffic engineering mode, include the traffic-engineering statement. Configuring OSPF passive traffic engineering mode enables the dynamic discovery of OSPF AS boundary routers.

Enable OSPF on an interface by including the interface statement at the [edit protocols (ospf | ospf3) area *area-id*] or the [edit routing-instances *routing-instance-name* protocols ospf area *area-id*] hierarchy levels. Disable it by including the disable statement, To prevent OSPF from running on an interface, include the passive statement. These three states are mutually exclusive.

Required Privilege Level

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

Related Topics

- disable (OSPF)
- Advertising Interface Addresses Without Running OSPF on page 479
- Configuring OSPF Passive Traffic Engineering Mode on page 479

peer-interface

Syntax

```
peer-interface interface-name {
    disable;
    dead-interval seconds;
    hello-interval seconds;
    retransmit-interval seconds;
    transit-delay seconds;
}
```

Hierarchy Level [edit logical-systems *logical-system-name* protocols ospf area *area-id*], [edit protocols ospf area *area-id*]

Release Information Statement introduced before JUNOS Release 7.4.

Description Configure a peer interface.

Options *interface-name*—Name of the peer interface. To configure all interfaces, you can specify all. For details about specifying interfaces, see the *JUNOS Network Interfaces Configuration Guide*.

The remaining statements are explained separately.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics

- Configuring OSPF Peer Interfaces on page 488

poll-interval

Syntax poll-interval *seconds*;

Hierarchy Level [edit logical-systems *logical-system-name* protocols (ospf | ospf3) area *area-id* interface *interface-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols (ospf | ospf3) area *area-id* interface *interface-name*],
[edit protocols (ospf | ospf3) area *area-id* interface *interface-name*],
[edit routing-instances *routing-instance-name* protocols (ospf | ospf3) area *area-id* interface *interface-name*]

Release Information Statement introduced before JUNOS Release 7.4.

Description For nonbroadcast interfaces only, specify how often the router sends hello packets out of the interface before it establishes adjacency with a neighbor.

Options *seconds*—Frequency at which to send hello packets.
Range: 1 through 255 seconds
Default: 120 seconds

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring an Interface on a Nonbroadcast, Multiaccess Network on page 455

preference

Syntax preference *preference*;

Hierarchy Level
[edit logical-systems *logical-system-name* protocols (ospf | ospf3)],
[edit logical-systems *logical-system-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast)],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols (ospf | ospf3)],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast)],
[edit protocols (ospf | ospf3)],
[edit protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast)],
[edit routing-instances *routing-instance-name* protocols (ospf | ospf3)],
[edit routing-instances *routing-instance-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast)]

Release Information
Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.
Support for the realm statement introduced in JUNOS Release 9.2.
Support for the realm statement introduced in JUNOS Release 9.2 for EX Series switches.

Description Set the route preference for OSPF internal routes.

Options *preference*—Preference value.
Range: 0 through 4,294,967,295 ($2^{32} - 1$)
Default: 10

Required Privilege Level
routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- external-preference
- Configuring Preference Values for OSPF Routes on page 465

prefix-export-limit

Syntax prefix-export-limit *number*;

Hierarchy Level
[edit logical-systems *logical-system-name* protocols (ospf | ospf3)],
[edit logical-systems *logical-system-name* protocols ospf topology (default | ipv4-multicast | *name*)],
[edit logical-systems *logical-system-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast)],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols (ospf | ospf3)],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ospf topology (default | ipv4-multicast | *name*)],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast)],
[edit protocols (ospf | ospf3)],
[edit protocols ospf topology (default | ipv4-multicast | *name*)],
[edit protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast)],
[edit routing-instances *routing-instance-name* protocols (ospf | ospf3)],
[edit routing-instances *routing-instance-name* protocols ospf topology (default | ipv4-multicast | *name*)],
[edit routing-instances *routing-instance-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast)]

Release Information
Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.
Support for Multitopology Routing introduced in JUNOS Release 9.0.
Support for Multitopology Routing introduced in JUNOS Release 9.0 for EX Series switches.
Support for the realm statement introduced in JUNOS Release 9.2.
Support for the realm statement introduced in JUNOS Release 9.2 for EX Series switches.

Description Configure a limit to the number of prefixes exported into OSPF.

Options *number*—Prefix limit.
Range: 0 through 4,294,967,295 ($2^{32} - 1$)
Default: None

Required Privilege Level
routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Limiting the Number of Prefixes Exported to OSPF on page 462
- Configuring a Prefix Export Limit for MT-OSPF on page 294

priority

Syntax priority *number*;

Hierarchy Level
[edit logical-systems *logical-system-name* protocols (ospf | ospf3) area *area-id* interface *interface-name*],
[edit logical-systems *logical-system-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast)] area *area-id* interface *interface-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols (ospf | ospf3) area *area-id* interface *interface-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast)] area *area-id* interface *interface-name*],
[edit protocols (ospf | ospf3) area *area-id* interface *interface-name*],
[edit protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast)] area *area-id* interface *interface-name*],
[edit routing-instances *routing-instance-name* protocols (ospf | ospf3) area *area-id* interface *interface-name*],
[edit routing-instances *routing-instance-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast)] area *area-id* interface *interface-name*]

Release Information
Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.
Support for the realm statement introduced in JUNOS Release 9.2.
Support for the realm statement introduced in JUNOS Release 9.2 for EX Series switches.

Description Specify the routing device's priority for becoming the designated routing devices. The routing device that has the highest priority value on the logical IP network or subnet becomes the network's designated router. You must configure at least one routing device on each logical IP network or subnet to be the designated router. You also should specify a routing device's priority for becoming the designated router on point-to-point interfaces.

Options
number—Routing device's priority for becoming the designated router. A priority value of 0 means that the routing device never becomes the designated router. A value of 1 means that the routing device has the least chance of becoming a designated router.
Range: 0 through 255
Default: 128

Required Privilege Level
routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- OSPF Designated Router Overview on page 439
- Configuring the Designated Router Priority for OSPF on page 462

realm

Syntax

```
realm (ipv4-unicast | ipv4-multicast | ipv6-unicast) {
    area area-id {
        interface interface-name;
    }
}
```

Hierarchy Level

[edit logical-systems *logical-system-name* protocols ospf3],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ospf3],
[edit protocols ospf3],
[edit routing-instances *routing-instance-name* protocols ospf3]

Release Information

Statement introduced in JUNOS Release 9.2.
Statement introduced in JUNOS Release 9.2 for EX Series switches.

Description

Configure OSPFv3 to advertise address families other than unicast IPv6. The JUNOS Software maps each address family you configure to a separate realm with its own set of neighbors and link-state database.

Options

ipv4-unicast—Configure a realm for IPv4 unicast routes.

ipv4-multicast—Configure a realm for IPv4 multicast routes.

ipv6-multicast—Configure a realm for IPv6 multicast routes.

Required Privilege Level

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

Related Topics

- Configuring Multiple Address Families for OSPFv3 on page 457

reference-bandwidth

Syntax reference-bandwidth *reference-bandwidth*;

Hierarchy Level
[edit logical-systems *logical-system-name* protocols (ospf | ospf3)],
[edit logical-systems *logical-system-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast)],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols (ospf | ospf3)],
[edit logical-systems *logical-system-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast)],
[edit protocols (ospf | ospf3)],
[edit protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast)],
[edit routing-instances *routing-instance-name* protocols (ospf | ospf3)],
[edit routing-instances *routing-instance-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast)]

Release Information
Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.
Support for the realm statement introduced in JUNOS Release 9.2.
Support for the realm statement introduced in JUNOS Release 9.2 for EX Series switches.

Description Set the reference bandwidth used in calculating the default interface cost. The cost is calculated using the following formula:

```
cost = ref-bandwidth/bandwidth
```

Options *ref-bandwidth*—Reference bandwidth, in bits per second.
Default: 100 Mbps (100,000,000 bits)
Range: 9600 through 1,000,000,000,000 bits

NOTE: The default behavior is to use the reference-bandwidth value to calculate the cost of OSPF interfaces. You can override this behavior for any OSPF interface by configuring a specific cost with the metric statement.

Required Privilege Level
routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- metric
- Configuring the Metric Value for OSPF Interfaces on page 464

retransmit-interval

Syntax retransmit-interval *seconds*;

Hierarchy Level
[edit logical-systems *logical-system-name* protocols ospf area *area-id* peer-interface *interface-name*],
[edit logical-systems *logical-system-name* protocols (ospf | ospf3) area *area-id* interface *interface-name*],
[edit logical-systems *logical-system-name* protocols (ospf | ospf3) area *area-id* virtual-link],
[edit logical-systems *logical-system-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast) area *area-id* interface *interface-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols (ospf | ospf3) area *area-id* interface *interface-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols (ospf | ospf3) area *area-id* virtual-link],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast) area *area-id* interface *interface-name*],
[edit protocols ospf area *area-id* peer-interface *interface-name*],
[edit protocols (ospf | ospf3) area *area-id* interface *interface-name*],
[edit protocols (ospf | ospf3) area *area-id* virtual-link],
[edit protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast) area *area-id* interface *interface-name*],
[edit routing-instances *routing-instance-name* protocols (ospf | ospf3) area *area-id* interface *interface-name*],
[edit routing-instances *routing-instance-name* protocols (ospf | ospf3) area *area-id* virtual-link],
[edit routing-instances *routing-instance-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast) area *area-id* interface *interface-name*]

Release Information
Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.
Support for the realm statement introduced in JUNOS Release 9.2.
Support for the realm statement introduced in JUNOS Release 9.2 for EX Series switches.

Description Specify how long the routing device waits to receive a link-state acknowledgment packet before retransmitting link-state advertisements to an interface's neighbors.

Options *seconds*—Interval to wait.
Range: 1 through 65,535 seconds
Default: 5 seconds

NOTE: You must configure link-state advertisement (LSA) retransmit intervals to be equal to or greater than 3 seconds to avoid triggering a retransmit trap, because the JUNOS Software delays LSA acknowledgments by up to 2 seconds.

Required Privilege Level
routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Controlling the LSA Retransmission Interval on page 467

rib-group

Syntax
rib-group *group-name*;

Hierarchy Level
[edit logical-systems *logical-system-name* protocols (ospf | ospf3)],
[edit logical-systems *logical-system-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast)],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols (ospf | ospf3)],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast)],
[edit protocols (ospf | ospf3)],
[edit protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast)],
[edit routing-instances *routing-instance-name* protocols (ospf | ospf3)],
[edit routing-instances *routing-instance-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast)]

Release Information
Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.
Support for the realm statement introduced in JUNOS Release 9.2.
Support for the realm statement introduced in JUNOS Release 9.2 for EX Series switches.

Description
Install routes learned from OSPF routing instances into routing tables in the OSPF routing table group.

Options
group-name—Name of the routing table group.

Required Privilege Level
routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- interface-routes
- rib-group
- Creating Routing Table Groups on page 114
- Configuring How Interface Routes Are Imported into Routing Tables on page 116
- Configuring BGP Routing Table Groups on page 772

route-type-community

Syntax route-type-community (iana | vendor);

Hierarchy Level [edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols (ospf | ospf3)],
[edit routing-instances *routing-instance-name* protocols (ospf | ospf3)]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Specify an extended community value to encode the OSPF route type. Each extended community is coded as an eight-octet value. This statement sets the most significant bit to either an IANA or vendor-specific route type.

Options iana—Encode a route type with the value 0x0306. This is the default value.

vendor—Encode the route type with the value 0x8000.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring OSPF Domain IDs for VPNs on page 266

secondary

Syntax secondary;

Hierarchy Level [edit logical-systems *logical-system-name* protocols ospf area *area-id* interface *interface-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ospf area *area-id* interface *interface-name*],
[edit protocols ospf area *area-id* interface *interface-name*],
[edit routing-instances *routing-instance-name* protocols ospf area *area-id* interface *interface-name*]

Release Information Statement introduced in JUNOS Release 9.2.

Description Configure an interface to belong to another OSPF area. A logical interface can be configured as primary interface only for one area. For any other area for which you configure the interface, you must configure it as a secondary interface.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- interface
- Configuring Multiarea Adjacency in OSPFv2 on page 457

sham-link

Syntax

```
sham-link {
    local address;
}
```

Hierarchy Level [edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ospf],
[edit routing-instances *routing-instance-name* protocols ospf]

Release Information Statement introduced before JUNOS Release 7.4.

Description Configure the local endpoint of a sham link.

Options local *address*—Local endpoint address.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration

Related Topics
- Configuring OSPF Sham Links on page 488

sham-link-remote

Syntax

```
sham-link-remote address {
    demand-circuit;
    ipsec-sa name;
    metric metric;
}
```

Hierarchy Level [edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ospf area *area-id*],
[edit routing-instances *routing-instance-name* protocols ospf area *area-id*]

Release Information Statement introduced before JUNOS Release 7.4.
Support for ipsec-sa statement added in JUNOS Release 8.3.

Description Configure the remote endpoint of a sham link.

The remaining statements are explained separately.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring OSPF Sham Links on page 488
- Configuring Authentication for OSPFv2 on page 458

shortcuts

Syntax

```
shortcuts;
    lsp-metric-into-summary;
}
```

Hierarchy Level

[edit logical-systems *logical-system-name* protocols (ospf | ospf3) traffic-engineering],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols (ospf | ospf3) traffic-engineering],
[edit protocols (ospf | ospf3) traffic-engineering],
[edit routing-instances *routing-instance-name* protocols (ospf | ospf3)traffic-engineering]

Release Information

Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.
Support for OSPFv3 (ospf3) introduced in JUNOS Release 9.4.
Support for OSPFv3 (ospf3) introduced in JUNOS Release 9.4 for EX Series switches.

Description

Configure OSPF to use MPLS label-switched paths (LSPs) as shortcut next hops. By default, shortcut routes calculated through OSPFv2 are installed in theinet.3 routing table, and shortcut routes calculated through OSPFv3 are installed in the inet6.3 routing table.

Required Privilege Level

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

Related Topics

- Enabling OSPF Traffic Engineering Support on page 481

simple-password

Syntax simple-password *key*;

Hierarchy Level
[edit logical-systems *logical-system-name* protocols ospf area *area-id* interface *interface-name* authentication],
[edit logical-systems *logical-system-name* protocols ospf area *area-id* virtual-link authentication],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ospf area *area-id* interface *interface-name* authentication],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ospf area *area-id* virtual-link authentication],
[edit protocols ospf area *area-id* interface *interface-name* authentication],
[edit protocols ospf area *area-id* virtual-link authentication],
[edit routing-instances *routing-instance-name* protocols ospf area *area-id* interface *interface-name* authentication],
[edit routing-instances *routing-instance-name* protocols ospf area *area-id* virtual-link authentication]

Release Information Statement introduced before JUNOS Release 7.4.

Description Configure a simple authentication key (password).

Options *key*—Password string.

Required Privilege Level
routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring Authentication for OSPFv2 on page 458

spf-options

Syntax

```
spf-options {
    delay milliseconds;
    holddown milliseconds;
    rapid-runs number;
}
```

Hierarchy Level

[edit logical-systems *logical-system-name* protocols (ospf | ospf3)],
[edit logical-systems *logical-system-name* protocols ospf topology (default | ipv4-multicast | *name*)],
[edit logical-systems *logical-system-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast)],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols (ospf | ospf3)],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ospf topology (default | ipv4-multicast | *name*)],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast)],
[edit protocols (ospf | ospf3)],
[edit protocols ospf topology (default | ipv4-multicast | *name*)],
[edit protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast)],
[edit routing-instances *routing-instance-name* protocols (ospf | ospf3)],
[edit routing-instances *routing-instance-name* protocols ospf topology (default | ipv4-multicast | *name*)],
[edit routing-instances *routing-instance-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast)]

Release Information

Statement introduced in JUNOS Release 8.5.
Statement introduced in JUNOS Release 9.0 for EX Series switches.
Support for Multitopology Routing introduced in JUNOS Release 9.0.
Support for Multitopology Routing introduced in JUNOS Release 9.0 for EX Series switches.
Support for the realm statement introduced in JUNOS Release 9.2.
Support for the realm statement introduced in JUNOS Release 9.2 for EX Series switches.

Description

Configure options for running the shortest-path-first (SPF) algorithm. You can configure a delay for when to run the SPF algorithm after a network topology change is detected, the maximum number of times the SPF algorithm can run in succession, and a hold-down interval after the SPF algorithm runs the maximum number of times.

Options

delay *milliseconds*—Time interval between the detection of a topology change and when the SPF algorithm runs.
Range: 50 through 8000 milliseconds
Default: 200 milliseconds

holddown *milliseconds*—Time interval to hold down, or wait before a subsequent SPF algorithm runs after the SPF algorithm has run the configured maximum number of times in succession.
Range: 2000 through 20,000 milliseconds

Default: 5000 milliseconds

rapid-runs *number*—Maximum number of times the SPF algorithm can run in succession. After the maximum is reached, the holddown interval begins.
Range: 1 through 5
Default: 3

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring SPF Options for OSPF on page 478
- Configuring Topologies and SPF Options for MT-OSPF on page 292

stub

Syntax stub <default-metric *metric*> <(no-summaries | summaries)>;

Hierarchy Level
[edit logical-systems *logical-system-name* protocols (ospf | ospf3) area *area-id*],
[edit logical-systems *logical-system-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast)],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols (ospf | ospf3) area *area-id*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast)],
[edit protocols (ospf | ospf3) area *area-id*],
[edit protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast)],
[edit routing-instances *routing-instance-name* protocols (ospf | ospf3) area *area-id*],
[edit routing-instances *routing-instance-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast)]

Release Information
Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.
Support for the realm statement introduced in JUNOS Release 9.2.
Support for the realm statement introduced in JUNOS Release 9.2 for EX Series switches.

Description Specify that this area not be flooded with AS external link-state advertisements (LSA)s. You must include the stub statement when configuring all routing devices that are in the stub area.

The backbone cannot be configured as a stub area.

You cannot configure an area to be both a stub area and a not-so-stubby area (NSSA).

Options
no-summaries—(Optional) Do not advertise routes into the stub area. If you include the default-metric option, only the default route is advertised.

summaries—(Optional) Flood summary LSAs into the stub area.

The remaining statement is explained separately.

Required Privilege Level
routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- nssa
- Configuring OSPF Stub Areas on page 450

summaries

Syntax (summaries | no-summaries);

Hierarchy Level
[edit logical-systems *logical-system-name* protocols (ospf | ospf3) area *area-id* nssa],
[edit logical-systems *logical-system-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast) area *area-id* nssa],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols (ospf | ospf3) area *area-id* nssa],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast) area *area-id* nssa],
[edit protocols (ospf | ospf3) area *area-id* nssa],
[edit protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast)] area *area-id* nssa],
[edit routing-instances *routing-instance-name* protocols (ospf | ospf3) area *area-id* nssa],
[edit routing-instances *routing-instance-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast) area *area-id* nssa]

Release Information
Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.
Support for the realm statement introduced in JUNOS Release 9.2.
Support for the realm statement introduced in JUNOS Release 9.2 for EX Series switches.

Description Configure whether or not area border routers advertise summary routes into an not-so-stubby area (NSSA):

- summaries—Flood summary link-state advertisements (LSAs) into the NSSA.
- no-summaries—Prevent area border routers from advertising summaries into an NSSA. If default-metric is configured for an NSSA, a Type 3 LSA is injected into the area by default.

Required Privilege Level
routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- nssa
- stub
- Configuring OSPF Not-So-Stubby Areas on page 451

te-metric

Syntax te-metric *metric*;

Hierarchy Level [edit logical-systems *logical-system-name* protocols ospf area *area-id* interface *interface-name*],
[edit protocols ospf area *area-id* interface *interface-name*]

Release Information Statement introduced before JUNOS Release 7.4.

Description Metric value used by traffic engineering for information injected into the traffic engineering database. The value of the traffic engineering metric does not affect normal OSPF forwarding.

Options *metric*—Metric value.
Range: 1 through 65,535
Default: Value of the IGP metric

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring the OSPF Metric Value Used for Traffic Engineering on page 484

traceoptions

Syntax

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

Hierarchy Level

[edit logical-systems *logical-system-name* protocols (ospf | ospf3)],
[edit logical-systems *logical-system-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast)],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols (ospf | ospf3)],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast)],
[edit protocols (ospf | ospf3)],
[edit protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast)],
[edit routing-instances *routing-instance-name* protocols (ospf | ospf3)],
[edit routing-instances *routing-instance-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast)]

Release Information

Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.
Support for the realm statement introduced in JUNOS Release 9.2.
Support for the realm statement introduced in JUNOS Release 9.2 for EX Series switches.

Description

Configure OSPF protocol-level tracing options.

To specify more than one tracing operation, include multiple flag statements.

Default

The default OSPF protocol-level tracing options are those inherited from the routing protocols traceoptions statement included at the [edit routing-options] hierarchy level.

Options

disable—(Optional) Disable the tracing operation. You can use this option to disable a single operation when you have defined a broad group of tracing operations, such as all.

file *filename*—Name of the file to receive the output of the tracing operation. Enclose the name within quotation marks. All files are placed in the directory /var/log. We recommend that you place OSPF tracing output in the file ospf-log.

files *number*—(Optional) Maximum number of trace files. When a trace file named *trace-file* reaches its maximum size, it is renamed *trace-file*.0, then *trace-file*.1, and so on, until the maximum number of trace files is reached. Then, the oldest trace file is overwritten.
If you specify a maximum number of files, you also must specify a maximum file size with the size option.
Range: 2 through 1000 files
Default: 10 files

flag—Tracing operation to perform. To specify more than one tracing operation, include multiple **flag** statements.

OSPF Tracing Flags

- **database-description**—Database description packets, which are used in synchronizing the OSPF and OSPFv3 topological database.
- **error**—OSPF and OSPFv3 error packets.
- **event**—OSPF and OSPFv3 state transitions.
- **flooding**—Link-state flooding packets.
- **graceful-restart**—Graceful-restart events.
- **hello**—Hello packets, which are used to establish neighbor adjacencies and to determine whether neighbors are reachable.
- **ldp-synchronization**—Synchronization events between OSPF and LDP
- **lsa-ack**—Link-state acknowledgment packets, which are used in synchronizing the OSPF topological database.
- **lsa-analysis**—Link-state analysis packets
- **lsa-request**—Link-state request packets, which are used in synchronizing the OSPF topological database.
- **lsa-update**—Link-state updates packets, which are used in synchronizing the OSPF topological database.
- **nsr-synchronization**—Nonstop routing synchronization events.
- **on-demand**—Trace demand circuit extensions.
- **packet-dump**—Content of selected packet types.
- **packets**—All OSPF packets.
- **spf**—Shortest-path-first (SPF) calculations.

Global Tracing Flags

- **all**—All tracing operations.
- **general**—A combination of the **normal** and **route** trace operations.
- **normal**—All normal operations.

  Default: If you do not specify this option, only unusual or abnormal operations are traced.
- **policy**—Policy operations and actions.
- **route**—Routing table changes.
- **state**—State transitions.
- **task**—Routing protocol task processing.
- **timer**—Routing protocol timer processing.

flag-modifier—(Optional) Modifier for the tracing flag. You can specify one or more of these modifiers:

- detail—Provide detailed trace information.
- receive—Packets being received.
- send—Packets being transmitted.

no-world-readable—(Optional) Prevent any user from reading the log file.

size *size*—(Optional) Maximum size of each trace file, in kilobytes (KB), megabytes (MB), or gigabytes (GB). When a trace file named *trace-file* reaches this size, it is renamed *trace-file*.0. When the *trace-file* again reaches its maximum size, *trace-file*.0 is renamed *trace-file*.1 and *trace-file* is renamed *trace-file*.0. This renaming scheme continues until the maximum number of trace files is reached. Then, the oldest trace file is overwritten.
If you specify a maximum file size, you also must specify a maximum number of trace files with the files option.
Syntax: *x*k to specify KB, *x*m to specify MB, or *x*g to specify GB
Range: 10 KB through the maximum file size supported on your system
Default: 128 KB

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

Required Privilege Level routing and trace—To view this statement in the configuration.
routing-control and trace-control—To add this statement to the configuration.

Related Topics
- Tracing OSPF Protocol Traffic on page 495

traffic-engineering

See the following sections:

- traffic-engineering (OSPF) on page 574
- traffic-engineering (Passive TE Mode) on page 575

traffic-engineering (OSPF)

Syntax

```
traffic-engineering {
    <advertise-unnumbered-interfaces>;
    <credibility-protocol-preference>;
    ignore-lsp-metrics;
    multicast-rpf-routes;
    no-topology;
    shortcuts {
        lsp-metric-into-summary;
    }
}
```

Hierarchy Level

[edit logical-systems *logical-system-name* protocols (ospf | ospf3)],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols (ospf | ospf3)],
[edit protocols (ospf | ospf3)],
[edit routing-instances *routing-instance-name* protocols (ospf | ospf3)]

Release Information

Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.
multicast-rpf-routes option introduced in JUNOS Release 7.5.
multicast-rpf-routes option introduced in JUNOS Release 9.0 for EX Series switches.
advertise-unnumbered-interfaces option introduced in JUNOS Release 8.5.
advertise-unnumbered-interfaces option introduced in JUNOS Release 9.0 for EX Series switches.
Support for OSPFv3 (ospf3) introduced in JUNOS Release 9.4.
Support for OSPFv3 (ospf3) introduced in JUNOS Release 9.0 for EX Series switches.
credibility-protocol-preference statement introduced in JUNOS Release 9.4.
credibility-protocol-preference statement introduced in JUNOS Release 9.4 for EX Series switches.

Description

Enable the OSPF traffic engineering features.

Default

Traffic engineering support is disabled.

Options

advertise-unnumbered-interfaces—(Optional) (OSPFv2 only) Include the link-local identifier in the link-local traffic-engineering link-state advertisement. You do not need to include this statement if RSVP is able to signal unnumbered interfaces as defined in RFC 3477.

credibility-protocol-preference—(Optional) (OSPFv2 only) Specify to use the configured preference value for OSPF routes to calculate the traffic engineering database credibility value used to select IGP routes. Use this statement to override the default behavior of having the traffic engineering database prefer IS-IS routes even if OSPF routes are configured a with a lower, that is, preferred, preference value.

multicast-rpf-routes—(Optional) (OSPFv2 only) Install routes for multicast RPF checks into the inet.2 routing table.

no-topology—(Optional) (OSPFv2 only) Disable the dissemination of the link-state topology information.

The remaining statements are explained separately.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Enabling OSPF Traffic Engineering Support on page 481

traffic-engineering (Passive TE Mode)

Syntax
```
traffic-engineering {
    remote-node-id address;
}
```

Hierarchy Level
[edit logical-systems *logical-system-name* protocols (ospf | ospf3) area *area-id* interface *interface-name* passive],
[edit logical-systems *logical-system-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast) area *area-id* interface *interface-name* passive],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols (ospf | ospf3) area *area-id* interface *interface-name* passive],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast) area *area-id* interface *interface-name* passive],
[edit protocols (ospf | ospf3) area *area-id* interface *interface-name* passive],
[edit protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast) area *area-id* interface *interface-name* passive],
[edit routing-instances *routing-instance-name* protocols (ospf | ospf3) area *area-id* interface *interface-name* passive],
[edit routing-instances *routing-instance-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast) area *area-id* interface *interface-name* passive]

Release Information Statement introduced in JUNOS Release 8.0.
Statement introduced in JUNOS Release 9.0 for EX Series switches.
Support for the realm statement introduced in JUNOS Release 9.2.
Support for the realm statement introduced in JUNOS Release 9.2 for EX Series switches.

Description Configure an interface in OSPF passive traffic engineering mode to enable dynamic discovery of OSPF AS boundary routers.

Default OSPF passive traffic-engineering mode is disabled.

Options remote-node-id *address*—The IP address at the far end of the inter-AS link.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring OSPF Passive Traffic Engineering Mode on page 479
- *JUNOS MPLS Applications Configuration Guide*

transit-delay

Syntax transit-delay *seconds*;

Hierarchy Level
[edit logical-systems *logical-system-name* protocols ospf area *area-id* peer-interface *interface-name*],
[edit logical-systems *logical-system-name* protocols (ospf | ospf3) area *area-id* interface *interface-name*],
[edit logical-systems *logical-system-name* protocols (ospf | ospf3) area *area-id* virtual-link],
[edit logical-systems *logical-system-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast) area *area-id* interface *interface-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ospf area *area-id* interface *interface-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ospf area *area-id* virtual-link],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast) area *area-id* interface *interface-name*],
[edit protocols ospf area *area-id* peer-interface *interface-name*],
[edit protocols (ospf | ospf3) area *area-id* interface *interface-name*],
[edit protocols (ospf | ospf3) area *area-id* virtual-link],
[edit protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast)] area *area-id* interface *interface-name*],
[edit routing-instances *routing-instance-name* protocols ospf area *area-id* interface *interface-name*],
[edit routing-instances *routing-instance-name* protocols ospf area *area-id* virtual-link],
[edit routing-instances *routing-instance-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast) area *area-id* interface *interface-name*]

Release Information
Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.
Support for the realm statement introduced in JUNOS Release 9.2.
Support for the realm statement introduced in JUNOS Release 9.2 for EX Series switches.

Description Set the estimated time required to transmit a link-state update on the interface. When calculating this time, make sure to account for transmission and propagation delays.

You should never have to modify the transit delay time.

Options *seconds*—Estimated time, in seconds.
Range: 1 through 65,535 seconds
Default: 1 second

Required Privilege Level
routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Specifying the Transit Delay on page 467

transmit-interval

Syntax transmit-interval *milliseconds*;

Hierarchy Level [edit logical-systems *logical-system-name* protocols (ospf | ospf3) area *area-id* interface *interface-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols (ospf | ospf3) area *area-id* interface *interface-name*],
[edit protocols (ospf | ospf3) area *area-id* interface *interface-name*],
[edit routing-instances *routing-instance-name* protocols (ospf | ospf3) area *area-id* interface *interface-name*]

Release Information Statement introduced before JUNOS Release 7.4.

Description Set the interval at which OSPF packets are transmitted on an interface.

Options *milliseconds*—Transmission interval, in milliseconds.
Range: 1 through 4,294,967 milliseconds
Default: 30 milliseconds

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Controlling the LSA Retransmission Interval on page 467

type-7

Syntax type-7;

Hierarchy Level
[edit logical-systems *logical-system-name* protocols (ospf | ospf3) area *area-id* nssa default-lsa],
[edit logical-systems *logical-system-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast) area *area-id* nssa default-lsa],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols (ospf | ospf3) area *area-id* nssa default-lsa],
[edit logical-systems *logical-system-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast) area *area-id* nssa default-lsa],
[edit protocols (ospf | ospf3) area *area-id* nssa default-lsa],
[edit protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast) area *area-id* nssa default-lsa],
[edit routing-instances *routing-instance-name* protocols (ospf | ospf3) area *area-id* nssa default-lsa],
[edit routing-instances *routing-instance-name* protocols ospf3 realm (ipv4-unicast | ipv4-multicast | ipv6-multicast) area *area-id* nssa default-lsa]

Release Information
Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.
Support for the realm statement introduced in JUNOS Release 9.2.
Support for the realm statement introduced in JUNOS Release 9.2 for EX Series switches.

Description Flood Type 7 default link-state advertisements (LSAs) if the no-summaries statement is configured.

By default, when the no-summaries statement is configured, a Type 3 LSA is injected into not-so-stubby areas (NSSAs) for JUNOS Release 5.0 and later. To support backward compatibility with earlier JUNOS releases, include the type-7 statement. This statement enables NSSA ABRs to advertise a Type 7 default LSA into the NSSA if you have also included the no-summaries statement in the configuration.

Required Privilege Level
routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring OSPF Not-So-Stubby Areas on page 451

virtual-link

Syntax

```
virtual-link neighbor-id router-id transit-area area-id {
    disable;
    authentication key <key-id identifier>;
    dead-interval seconds;
    hello-interval seconds;
    ipsec-sa name;
    retransmit-interval seconds;
    transit-delay seconds;
}
```

Hierarchy Level

[edit logical-systems *logical-system-name* protocols (ospf | ospf3) area *area-id*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ospf area *area-id*],
[edit protocols (ospf | ospf3) area *area-id*],
[edit routing-instances *routing-instance-name* protocols ospf area *area-id*]

Release Information

Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description

For backbone areas only, create a virtual link to use in place of an actual physical link. All area border routers and other routing devices on the backbone must be contiguous. If this is not possible and there is a break in OSPF connectivity, use virtual links to create connectivity to the OSPF backbone. When configuring virtual links, you must configure links on the two routing devices that form the end points of the link, and both these two routing devices must be area border routers. You cannot configure links through stub areas.

Options

neighbor-id *router-id*—IP address of the routing device at the remote end of the virtual link.

transit-area *area-id*—Area identifier of the area through which the virtual link transits. Virtual links are not allowed to transit the backbone area.

The remaining statements are explained separately.

Required Privilege Level

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

Related Topics

- Configuring OSPF Virtual Links on page 452

Chapter 20

Introduction to RIP

This chapter discusses the following topics that provide background information about RIP:

- RIP Overview on page 581
- RIP Standards on page 582

RIP Overview

RIP is an interior gateway protocol (IGP) that uses a distance-vector algorithm to determine the best route to a destination, using the hop count as the metric.

This section discusses the following topics:

- RIP Protocol Overview on page 581
- RIP Packets on page 582

RIP Protocol Overview

The RIP IGP uses the Bellman-Ford, or *distance-vector*, algorithm to determine the best route to a destination. RIP uses the hop count as the metric. RIP allows hosts and routers to exchange information for computing routes through an IP-based network. RIP is intended to be used as an IGP in reasonably homogeneous networks of moderate size.

The JUNOS Software supports RIP versions 1 and 2.

NOTE: RIP is not supported for multipoint interfaces.

RIP version 1 packets contain the minimal information necessary to route packets through a network. However, this version of RIP does not support authentication or subnetting.

RIP uses User Datagram Protocol (UDP) port 520.

RIP has the following architectural limitations:

- The longest network path cannot exceed 15 hops (assuming that each network, or hop, has a cost of 1).

- RIP depends on counting to infinity to resolve certain unusual situations—When the network consists of several hundred routers, and when a routing loop has formed, the amount of time and network bandwidth required to resolve a next hop might be great.
- RIP uses only a fixed metric to select a route. Other IGPs use additional parameters, such as measured delay, reliability, and load.

RIP Packets

RIP packets contain the following fields:

- Command—Indicates whether the packet is a request or response message. Request messages seek information for the router's routing table. Response messages are sent periodically and also when a request message is received. Periodic response messages are called *update messages*. Update messages contain the command and version fields and 25 destinations (by default), each of which includes the destination IP address and the metric to reach that destination.
- Version number—Version of RIP that the originating router is running.
- Address family identifier—Address family used by the originating router. The family is always IP.
- Address—IP address included in the packet.
- Metric—Value of the metric advertised for the address.
- Mask—Mask associated with the IP address (RIP version 2 only).
- Next hop—IP address of the next-hop router (RIP version 2 only).

RIP Standards

RIP is defined in the following documents:

- RFC 1058, *Routing Information Protocol*
- RFC 2082, *RIP-2 MD-5 Authentication*
- RFC 2453, *RIP Version 2*

To access Internet Requests for Comments (RFCs) and drafts, go to the Internet Engineering Task Force (IETF) Web site at http://www.ietf.org.

Chapter 21

RIP Configuration Guidelines

This chapter discusses the following topics:

- Configuring RIP on page 583
- Minimum RIP Configuration on page 585
- Overview of RIP Global Properties on page 586
- Overview of RIP Neighbor Properties on page 586
- Configuring Authentication for RIP on page 587
- Configuring BFD for RIP on page 588
- Overview of BFD Authentication for RIP on page 590
- Configuring BFD Authentication for RIP on page 592
- Accepting RIP Packets with Nonzero Values in Reserved Fields on page 595
- Applying Policies to RIP Routes Imported from Neighbors on page 596
- Configuring the Number of Route Entries in RIP Update Messages on page 596
- Configuring the Metric Value Added to Imported RIP Routes on page 596
- Configuring RIP Update Messages on page 597
- Configuring Routing Table Groups for RIP on page 597
- Configuring RIP Timers on page 597
- Configuring Group-Specific RIP Properties on page 598
- Configuring Graceful Restart for RIP on page 600
- Disabling Strict Address Checking for RIP Messages on page 600
- Tracing RIP Protocol Traffic on page 601
- Example: Configuring RIP on page 603

Configuring RIP

To configure RIP, you include the following statements:

```
protocols {
    rip {
        any-sender;
        authentication-key password;
        authentication-type type;
        (check-zero | no-check-zero);
```

```
graceful-restart {
   disable;
   restart-time seconds;
}
holddown seconds;
import [ policy-names ];
message-size number;
metric-in metric;
receive receive-options;
rib-group group-name;
route-timeout seconds;
send send-options;
update-interval seconds;
traceoptions {
   file filename <files number> <size size> <world-readable | no-world-readable>;
   flag flag <flag-modifier> <disable>;
}
group group-name {
   bfd-liveness-detection {
      authentication {
         algorithm algorithm-name;
         key-chain key-chain-name;
         loose-check;
      }
      detection-time {
         threshold milliseconds;
      }
      minimum-interval milliseconds;
      minimum-receive-interval milliseconds;
      multiplier number;
      no-adaptation;
      transmit-interval {
         threshold milliseconds;
         minimum-interval milliseconds;
      }
      version (1 | automatic);
   }
   export [ policy-names ];
   metric-out metric;
   preference number;
   route-timeout seconds;
   update-interval seconds;
   neighbor neighbor-name {
      authentication-key password;
      authentication-type type;
      bfd-liveness-detection {
         authentication {
            algorithm algorithm-name;
            key-chain key-chain-name;
            loose-check;
         }
         detection-time {
            threshold milliseconds;
         }
         minimum-interval milliseconds;
         minimum-receive-interval milliseconds;
```

```
            multiplier number;
            transmit-interval {
               threshold milliseconds;
               minimum-interval milliseconds;
            }
            version (1 | automatic);
         }
         (check-zero | no-check-zero);
         import [ policy-names ];
         message-size number;
         metric-in metric;
         metric-out metric;
         receive receive-options;
         route-timeout seconds;
         send send-options;
         update-interval seconds;
      }
   }
}
```

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for these statements.

By default, RIP is disabled.

To have a router exchange routes with other routers, you must configure RIP groups and neighbors. RIP routes received from routers not configured as RIP neighbors are ignored. Likewise, RIP routes are advertised only to routers configured as RIP neighbors, with an appropriate RIP export policy applied.

Minimum RIP Configuration

For a routing device to accept RIP routes, you must include at least the rip, group, and neighbor statements. Routes received from routing devices that are not configured as neighbors are ignored. All other RIP configuration statements are optional. This minimum configuration defines one neighbor. Include one neighbor statement for each interface on which you want to receive routes. The local routing device imports all routes by default from this neighbor and does not advertise routes. The routing device can receive both version 1 and version 2 update messages, with 25 route entries per message. For routing instances, include the statements at the [edit routing-instances *routing-instance-name* protocols rip] hierarchy level.

```
protocols {
   rip {
      group group-name {
         neighbor interface-name {
         }
      }
   }
}
```

NOTE: When you configure RIP on an interface, you must also include the family inet statement at the [edit interfaces *interface-name* unit *logical-unit-number*] hierarchy level. For more information about the family inet statement, see the *JUNOS Network Interfaces Configuration Guide*.

Overview of RIP Global Properties

To define RIP global properties, which apply to all RIP neighbors, include one or more of the following statements.

```
authentication-key password;
authentication-type type;
(check-zero | no-check-zero);
import [ policy-names ];
message-size number;
metric-in metric;
receive receive-options;
rib-group group-name;
send send-options;
```

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for these statements.

For more information about configuring RIP global properties, see the following topics:

- Configuring Authentication for RIP on page 587
- Accepting RIP Packets with Nonzero Values in Reserved Fields on page 595
- Applying Policies to RIP Routes Imported from Neighbors on page 596
- Configuring the Number of Route Entries in RIP Update Messages on page 596
- Configuring the Metric Value Added to Imported RIP Routes on page 596
- Configuring RIP Update Messages on page 597
- Configuring Routing Table Groups for RIP on page 597

Overview of RIP Neighbor Properties

To define neighbor-specific properties, include one or more of the following statements.

```
neighbor neighbor-name {
    authentication-key password;
    authentication-type type;
    bfd-liveness-detection {
        authentication {
            algorithm algorithm-name;
            key-chain key-chain-name;
            loose-check;
```

```
        }
        detection-time {
            threshold milliseconds;
        }
        minimum-interval milliseconds;
        minimum-receive-interval milliseconds;
        transmit-interval {
            threshold milliseconds;
            minimum-interval milliseconds;
        }
        multiplier number;
        version (0 | 1 | automatic);
    }
    (check-zero | no-check-zero);
    import [ policy-names ];
    message-size number;
    metric-in metric;
    receive receive-options;
    send send-options;
}
```

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for these statements.

For more information about configuring RIP neighbor properties, see the following topics:

- Configuring Authentication for RIP on page 587
- Configuring BFD for RIP on page 588
- Accepting RIP Packets with Nonzero Values in Reserved Fields on page 595
- Applying Policies to RIP Routes Imported from Neighbors on page 596
- Configuring the Number of Route Entries in RIP Update Messages on page 596
- Configuring the Metric Value Added to Imported RIP Routes on page 596
- Configuring RIP Update Messages on page 597

Configuring Authentication for RIP

You can configure the router to authenticate RIP route queries. By default, authentication is disabled. You can use the following authentication method:

- Simple authentication—Uses a text password that is included in the transmitted packet. The receiving router uses an authentication key (password) to verify the packet.
- MD5 authentication—Creates an encoded checksum that is included in the transmitted packet. The receiving router uses an authentication key (password) to verify the packet's MD5 checksum.

To enable authentication and specify an authentication method and password, include the authentication-key and authentication-type statements:

```
authentication-key password;
authentication-type type;
```

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for these statements.

The password can be up to 16 contiguous characters and can include any ASCII strings.

Configuring BFD for RIP

The Bidirectional Forwarding Detection (BFD) protocol is a simple hello mechanism that detects failures in a network. Hello packets are sent at a specified, regular interval. A neighbor failure is detected when the routing device stops receiving a reply after a specified interval. BFD works with a wide variety of network environments and topologies. BFD failure detection times are shorter than RIP detection times, providing faster reaction times to various kinds of failures in the network. These timers are also adaptive. For example, a timer can adapt to a higher value if the adjacency fails, or a neighbor can negotiate a higher value for a timer than the one configured.

NOTE: To enable BFD for RIP, both sides of the connection must receive an update message from the peer. By default, RIP does not export any routes. Therefore you must enable update messages to be sent by configuring an export policy for routes before a BFD session is triggered.

To enable failure detection, include the bfd-liveness-detection statement:

```
bfd-liveness-detection {
    detection-time {
        threshold milliseconds;
    }
    minimum-interval milliseconds;
    minimum-receive-interval milliseconds;
    multiplier number;
    no-adaptation;
    transmit-interval {
        threshold milliseconds;
        minimum-interval milliseconds;
    }
    version (1 | automatic);
}
```

To specify the threshold for the adaptation of the detection time, include the threshold statement:

```
detection-time {
    threshold milliseconds;
}
```

When the BFD session detection time adapts to a value equal to or higher than the threshold, a single trap and a system log message are sent.

To specify the minimum transmit and receive interval for failure detection, include the minimum-interval statement:

```
minimum-interval milliseconds;
```

This value represents the minimum interval at which the local routing device transmits hello packets as well as the minimum interval at which the routing device expects to receive a reply from a neighbor with which it has established a BFD session. You can configure a value in the range from 1 through 255,000 milliseconds. You can also specify the minimum transmit and receive intervals separately.

NOTE: Specifying an interval less than 300 ms can cause undesired BFD flapping.

To specify only the minimum receive intervals for failure detection, include the minimum-receive-interval statement:

```
minimum-receive-interval milliseconds;
```

This value represents the minimum interval at which the local routing device expects to receive a reply from a neighbor with which it has established a BFD session. You can configure a value in the range from 1 through 255,00 milliseconds.

To specify the number of hello packets not received by a neighbor that causes the originating interface to be declared down, include the multiplier statement:

```
multiplier number;
```

The default is 3, and you can configure a value in the range from 1 through 255.

To specify only the minimum transmit interval for failure detection, include the minimum-interval statement:

```
transmit-interval {
    minimum-interval milliseconds;
}
```

This value represents the minimum interval at which the local routing device transmits hello packets to the neighbor with which it has established a BFD session. You can configure a value in the range from 1 through 255,000 milliseconds.

To specify the threshold for detecting the adaptation of the transmit interval, include the threshold statement:

```
transmit-interval {
    threshold milliseconds;
}
```

The threshold value must be greater than the transmit interval.

To specify the BFD version used for detection, include the version statement:

```
version (1 | automatic);
```

The default is to have the version detected automatically.

You can trace BFD operations by including the traceoptions statement at the [edit protocols bfd] hierarchy level. For more information, see “Tracing BFD Protocol Traffic” on page 78.

In JUNOS Release 9.0 and later, you can configure BFD sessions not to adapt to changing network conditions. To disable BFD adaptation, include the no-adaptation statement:

```
no-adaptation;
```

NOTE: We recommend that you not disable BFD adaptation unless it is preferable not to have BFD adaptation enabled in your network.

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for these statements.

Overview of BFD Authentication for RIP

BFD enables rapid detection of communication failures between adjacent systems. By default, authentication for BFD sessions is disabled. However, when running BFD over Network Layer protocols, the risk of service attacks can be significant. We strongly recommend using authentication if you are running BFD over multiple hops or through insecure tunnels. Beginning with JUNOS Release 9.6, the JUNOS Software supports authentication for BFD sessions running over RIP. BFD authentication is only supported in the domestic image and is not available in the export image.

You authenticate BFD sessions by specifying an authentication algorithm and keychain, and then associating that configuration information with a security authentication keychain using the keychain name.

The following sections describe the supported authentication algorithms, security keychains, and the level of authentication that can be configured:

- BFD Authentication Algorithms on page 590
- Security Authentication Keychains on page 591
- Strict Versus Loose Authentication on page 591

BFD Authentication Algorithms

JUNOS Software supports the following algorithms for BFD authentication:

- simple-password—Plain-text password. One to 16 bytes of plain text are used to authenticate the BFD session. One or more passwords may be configured. This method is the least secure and should be used only when BFD sessions are not subject to packet interception.
- keyed-md5—Keyed Message Digest 5 hash algorithm for sessions with transmit and receive intervals greater than 100 ms. To authenticate the BFD session, keyed MD5 uses one or more secret keys (generated by the algorithm) and a

sequence number that is updated periodically. With this method, packets are accepted at the receiving end of the session if one of the keys matches and the sequence number is greater than or equal to the last sequence number received. Although more secure than a simple password, this method is vulnerable to replay attacks. Increasing the rate at which the sequence number is updated can reduce this risk.

- meticulous-keyed-md5—Meticulous keyed Message Digest 5 hash algorithm. This method works in the same manner as keyed MD5, but the sequence number is updated with every packet. Although more secure than keyed MD5 and simple passwords, this method may take additional time to authenticate the session.
- keyed-sha-1—Keyed Secure Hash Algorithm I for sessions with transmit and receive intervals greater than 100 ms. To authenticate the BFD session, keyed SHA uses one or more secret keys (generated by the algorithm) and a sequence number that is updated periodically. The key is not carried within the packets. With this method, packets are accepted at the receiving end of the session if one of the keys matches and the sequence number is greater than the last sequence number received.
- meticulous-keyed-sha-1—Meticulous keyed Secure Hash Algorithm I. This method works in the same manner as keyed SHA, but the sequence number is updated with every packet. Although more secure than keyed SHA and simple passwords, this method may take additional time to authenticate the session.

NOTE: Nonstop active routing (NSR) is not supported with meticulous-keyed-md5 and meticulous-keyed-sha-1 authentication algorithms. BFD sessions using these algortihms may go down after a switchover.

Security Authentication Keychains

The security authentication keychain defines the authentication attributes used for authentication key updates. When the security authentication keychain is configured and associated with a protocol through the keychain name, authentication key updates can occur without interrupting routing and signaling protocols.

The authentication keychain contains one or more keychains. Each keychain contains one or more keys. Each key holds the secret data and the time at which the key becomes valid. The algorithm and keychain must be configured on both ends of the BFD session, and they must match. Any mismatch in configuration prevents the BFD session from being created.

BFD allows multiple clients per session, and each client can have its own keychain and algorithm defined. To avoid confusion, we recommend specifying only one security authentication keychain.

Strict Versus Loose Authentication

By default, strict authentication is enabled and authentication is checked at both ends of each BFD session. Optionally, to smooth migration from nonauthenticated sessions to authenticated sessions, you can configure *loose checking*. When loose

checking is configured, packets are accepted without authentication being checked at each end of the session. This feature is intended for transitional periods only.

Related Topics

- Configuring BFD Authentication for RIP on page 592
- bfd-liveness-detection
- authentication-key-chains statement in the *JUNOS System Basics Configuration Guide*
- show bfd session command in the *JUNOS Routing Protocols and Policies Command Reference*
- Configuring BFD for RIP on page 588

Configuring BFD Authentication for RIP

Beginning with JUNOS Release 9.6, you can configure authentication for BFD sessions running over RIP. Only three steps are needed to configure authentication on a BFD session:

1. Specify the BFD authentication algorithm for the RIP protocol.
2. Associate the authentication keychain with the RIP protocol.
3. Configure the related security authentication keychain.

The following sections provide instructions for configuring and viewing BFD authentication on RIP:

- Configuring BFD Authentication Parameters on page 592
- Viewing Authentication Information for BFD Sessions on page 594

Configuring BFD Authentication Parameters

BFD authentication can be configured for the entire RIP protocol, or a specific RIP group, neighbor, or routing instance.

To configure BFD authentication:

1. Specify the algorithm (keyed-md5, keyed-sha-1, meticulous-keyed-md5, meticulous-keyed-sha-1, or simple-password) to use.

```
[edit]
user@host# set protocols rip bfd-liveness-detection authentication algorithm
   keyed-sha-1
user@host# set protocols rip group rip-gr2 bfd-liveness-detection authentication
   algorithm keyed-sha-1
user@host# set protocols rip group rip-gr2 neighbor 10.10.32.7
   bfd-liveness-detection authentication algorithm keyed-sha-1
```

NOTE: Nonstop active routing (NSR) is not supported with meticulous-keyed-md5 and meticulous-keyed-sha-1 authentication algorithms. BFD sessions using these algortihms may go down after a switchover.

2. Specify the keychain to be used to associate BFD sessions on RIP with the unique security authentication keychain attributes. The keychain you specify must match a keychain name configured at the [edit security authentication key-chains] hierarchy level.

```
[edit]
user@host# set protocols rip bfd-liveness-detection authentication keychain
   bfd-rip
user@host# set protocols rip group rip-gr2 bfd-liveness-detection authentication
   keychain bfd-rip
user@host# set protocols rip group rip-gr2 neighbor 10.10.32.7
   bfd-liveness-detection authentication keychain bfd-rip
```

NOTE: The algorithm and keychain must be configured on both ends of the BFD session, and they must match. Any mismatch in configuration prevents the BFD session from being created.

3. Specify the unique security authentication information for BFD sessions:
   - The matching *key-chain-name* as specified in Step 2.
   - At least one *key*, a unique integer between 0 and 63. Creating multiple keys allows multiple clients to use the BFD session.
   - The *secret-data* used to allow access to the session.
   - The time at which the authentication key becomes active, *yyyy-mm-dd.hh:mm:ss*.

```
[edit security]
user@host# authentication-key-chains key-chain bfd-bgp key 53 secret
   $9$ggaJDmPQ6/tJgF/AtREVsyPsnCtUHm start-time 2009-06-14.10:00:00
```

4. (Optional) Specify loose authentication checking if you are transitioning from nonauthenticated sessions to authenticated sessions.

```
[edit]
user@host> set protocols rip bfd-liveness-detection authentication loose-check
user@host> set protocols rip group rip-gr2 bfd-liveness-detection authentication
   loose-check
user@host> set protocols rip group rip-gr2 neighbor 10.10.32.7
   bfd-liveness-detection authentication loose-check
```

5. (Optional) View your configuration using the show bfd session detail or show bfd session extensive command.
6. Repeat these steps to configure the other end of the BFD session.

NOTE: BFD authentication is only supported in the domestic image and is not available in the export image.

Viewing Authentication Information for BFD Sessions

You can view the existing BFD authentication configuration using the show bfd session detail and show bfd session extensive commands.

The following example shows BFD authentication configured for the rip-gr2 BGP group. It specifies the keyed SHA-1 authentication algorithm and a keychain name of bfd-rip. The authentication keychain is configured with two keys. Key 1 contains the secret data "9ggaJDmPQ6/tJgF/AtREVsyPsnCtUHm" and a start time of June 1, 2009 at 9:46:02 AM PST. Key 2 contains the secret data "9a5jiKW9I.reP38ny.TszF2/9" and a start time of June 1, 2009 at 3:29:20 PM PST.

```
[edit protocols rip]
group rip-gr2 {
   bfd-liveness-detection {
      authentication {
         algorithm keyed-sha-1;
         key-chain bfd-rip;
      }
   }
}
[edit security]
authentication key-chains {
   key-chain bfd-rip {
      key 1 {
         secret "$9$ggaJDmPQ6/tJgF/AtREVsyPsnCtUHm";
         start-time "2009-6-1.09:46:02 -0700";
      }
      key 2 {
         secret "$9$a5jiKW9I.reP38ny.TszF2/9";
         start-time "2009-6-1.15:29:20 -0700";
      }
   }
}
```

If you commit these updates to your configuration, you would see output similar to the following. In the output for the show bfd sessions detail command, Authenticate is displayed to indicate that BFD authentication is configured. For more information about the configuration, use the show bfd sessions extensive command. The output for this command provides the keychain name, the authentication algorithm and mode for each client in the session, and the overall BFD authentication configuration status, keychain name, and authentication algorithm and mode.

show bfd sessions detail

```
user@host# show bfd session detail

                                                Detect   Transmit
Address                 State     Interface     Time     Interval  Multiplier
50.0.0.2                Up        ge-0/1/5.0    0.900     0.300        3
 Client RIP, TX interval 0.300, RX interval 0.300, Authenticate
```

```
 Session up time 3d 00:34
 Local diagnostic None, remote diagnostic NbrSignal
 Remote state Up, version 1
 Replicated
```

show bfd sessions extensive

```
user@host# show bfd session extensive
                                                  Detect   Transmit
Address                 State     Interface       Time     Interval  Multiplier
50.0.0.2                Up        ge-0/1/5.0      0.900     0.300        3
 Client RIP, TX interval 0.300, RX interval 0.300, Authenticate
        keychain bfd-rip, algo keyed-sha-1, mode strict
 Session up time 00:04:42
 Local diagnostic None, remote diagnostic NbrSignal
 Remote state Up, version 1
 Replicated
 Min async interval 0.300, min slow interval 1.000
 Adaptive async TX interval 0.300, RX interval 0.300
 Local min TX interval 0.300, minimum RX interval 0.300, multiplier 3
 Remote min TX interval 0.300, min RX interval 0.300, multiplier 3
 Local discriminator 2, remote discriminator 2
 Echo mode disabled/inactive
 Authentication enabled/active, keychain bfd-rip, algo keyed-sha-1, mode strict
```

Related Topics

- Overview of BFD Authentication for RIP on page 590
- bfd-liveness-detection
- authentication-key-chains statement in the *JUNOS System Basics Configuration Guide*
- show bfd session command in the *JUNOS Routing Protocols and Policies Command Reference*
- Configuring BFD for RIP on page 588

Accepting RIP Packets with Nonzero Values in Reserved Fields

Some of the reserved fields in RIP version 1 packets must be zero, while in RIP version 2 packets most of these reserved fields can contain nonzero values. By default, RIP discards version 1 packets that have nonzero values in the reserved fields and version 2 packets that have nonzero values in the fields that must be zero. This default behavior implements the RIP version 1 and version 2 specifications.

If you find that you are receiving RIP version 1 packets with nonzero values in the reserved fields or RIP version 2 packets with nonzero values in the fields that must be zero, you can configure RIP to receive these packets in spite of the fact that they are being sent in violation of the specifications in RFC 1058 and RFC 2453. To receive packets whose reserved fields are nonzero, include the no-check-zero statement:

```
no-check-zero;
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

Applying Policies to RIP Routes Imported from Neighbors

To filter routes being imported by the local routing device from its neighbors, include the import statement and list the names of one or more policies to be evaluated. If you specify more than one policy, they are evaluated in order (first to last) and the first matching policy is applied to the route. If no match is found, the local routing device does not import any routes.

```
import [ policy-names ];
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

For more information about creating policies, see the *JUNOS Policy Framework Configuration Guide*.

Configuring the Number of Route Entries in RIP Update Messages

By default, RIP includes 25 route entries in each update message. To change the number of route entries in an update message, include the message-size statement:

```
message-size number;
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement

NOTE: To ensure interoperability with routers from other vendors, do not change the default number of route entries in a RIP update message.

Configuring the Metric Value Added to Imported RIP Routes

By default, RIP imports routes from the neighbors configured with the neighbor statement. These routes include those learned from RIP as well as those learned from other protocols. By default, the current route metric of routes that RIP imports from its neighbors is incremented by 1.

To change the default metric to be added to incoming routes, include the metric-in statement:

```
metric-in metric;
```

metric can be a value from 1 through 16.

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

Related Topics ■ neighbor

Configuring RIP Update Messages

You can configure whether the RIP update messages conform to RIP version 1 only, to RIP version 2 only, or to both versions. You can also disable the sending or receiving of update messages. To configure the sending and receiving of update messages, include the receive and send statements:

```
receive receive-options;
send send-options;
```

For a list of hierarchy levels at which you can include these statements and a list of the valid options, see the statement summary sections for these statements.

Configuring Routing Table Groups for RIP

You can install routes learned through RIP into multiple routing tables by configuring a routing table group. RIP routes are installed into each routing table that belongs to that routing table group. To configure a routing table group for RIP routes, include the rib-group statement:

```
rib-group group-name;
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

Configuring RIP Timers

You can configure various timers for RIP.

RIP routes expire when either a route timeout limit is met or a route metric reaches infinity, and the route is no longer valid. However, the expired route is retained in the routing table for a time period so that neighbors can be notified that the route has been dropped. This time period is set by configuring the hold-down timer. Upon expiration of the hold-down timer, the route is removed from the routing table.

To configure the hold-down timer for RIP, include the holddown statement:

```
holddown seconds;
```

seconds can be a value from 10 through 180. The default value is 120 seconds.

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

You can set a route timeout interval. If a route is not refreshed after being installed into the routing table by the specified time interval, the route is removed from the routing table.

To configure the route timeout for RIP, include the route-timeout statement:

route-timeout *seconds*;

seconds can be a value from 30 through 360. The default value is 180 seconds.

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

You can set an update time interval to periodically send out routes learned by RIP to neighbors.

To configure the update time interval, include the update-interval statement:

update-interval *seconds*;

seconds can be a value from 10 through 60. The default value is 30 seconds.

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

Configuring Group-Specific RIP Properties

You can group together neighbors that share the same export policy and export metric defaults. You configure group-specific RIP properties by including the group statement at the [edit protocols rip] hierarchy level. Each group must contain at least one neighbor. You should create a group for every export policy you have. To configure neighbors, see "Overview of RIP Neighbor Properties" on page 586.

```
[edit protocols rip]
group group-name {
    bfd-liveness-detection {
        authentication {
            algorithm algorithm-name;
            key-chain key-chain-name;
            loose-check;
        }
        detection-time {
            threshold milliseconds;
        }
        minimum-interval milliseconds;
        minimum-receive-interval milliseconds;
        transmit-interval {
            threshold milliseconds;
            minimum-interval milliseconds;
        }
        multiplier number;
        version (0 | 1 | automatic);
    }
    export [ policy-names ];
    preference number;
    metric-out metric;
    neighbor neighbor-options;
}
```

This section discusses the following tasks:

- Applying Policies to Routes Exported by RIP on page 599
- Configuring the Default Preference Value for RIP on page 599
- Configuring the Metric for Routes Exported by RIP on page 600

Applying Policies to Routes Exported by RIP

By default, RIP does not export routes it has learned to its neighbors. To enable RIP to export routes, apply one or more export policies. To apply export policies and to filter routes being exported from the local routing device to its neighbors, include the export statement and list the name of the policy to be evaluated:

```
export [ policy-names ];
```

To configure export policy globally for all RIP neighbors, include the export statement.

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

You can define one or more export policies. If no routes match the policies, the local routing device does not export any routes to its neighbors. Export policies override any metric values determined through calculations involving the values configured with the metric-in and metric-out statements (discussed in "Configuring the Metric Value Added to Imported RIP Routes" on page 596 and "Configuring the Metric for Routes Exported by RIP" on page 600, respectively).

NOTE: The export policy on RIP does not support manipulating routing information of the next hop.

For more information about creating policies, see the *JUNOS Policy Framework Configuration Guide*.

Configuring the Default Preference Value for RIP

By default, the JUNOS Software assigns a preference of 100 to routes that originate from RIP. When the JUNOS Software determines a route's preference to become the active route, the software selects the route with the lowest preference and installs this route into the forwarding table. (For more information about preferences, see "Route Preferences Overview" on page 6.)

To modify the default RIP preference value, include the preference statement:

```
preference preference;
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

preference can be a value from 0 through 4,294,967,295 ($2^{32} - 1$).

Configuring the Metric for Routes Exported by RIP

If you have included the export statement, RIP exports routes it has learned to the neighbors configured by including the neighbor statement. For more information about those statements, see “Applying Policies to Routes Exported by RIP” on page 599 “Overview of RIP Neighbor Properties” on page 586.

The metric associated with a RIP route (unless modified by an export policy) is the normal RIP metric. For example, a RIP route with a metric of 5 learned from a neighbor configured with a metric-in value of 2 is advertised with a combined metric of 7 when advertised to RIP neighbors in the same group. However, if this route was learned from a RIP neighbor in a different group or from a different protocol, the route is advertised with the metric value configured for that group with the metric-out statement. The default value for the metric-out statement is 1.

The metric for a route may be modified with an export policy. That metric is seen when the route is exported to the next hop.

To increase the metric for routes advertised outside a group, include the metric-out statement:

```
metric-out metric;
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

Configuring Graceful Restart for RIP

Graceful restart is disabled by default. You can globally enable graceful restart for all routing protocols at the [edit routing-options] hierarchy level.

You can configure graceful restart parameters specifically for RIP. To do this, include the graceful-restart statement:

```
graceful-restart {
   restart-time seconds;
}
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

To disable graceful restart for RIP, specify the disable statement. To configure a time period for the restart to finish, specify the restart-time statement.

Disabling Strict Address Checking for RIP Messages

If the sender of a RIP message does not belong to the subnet of the interface, the message is discarded. This situation may cause problems with dropped packets when RIP is running on point-to-point interfaces, or when the addresses on the interfaces do not fall in the same subnet. You can resolve this by disabling strict address checks on the RIP traffic.

To disable strict address checks, include the any-sender statement:

```
any-sender;
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

NOTE: The any-sender statement is supported only for peer-to-peer interfaces.

Tracing RIP Protocol Traffic

You can trace various types of RIP protocol traffic to help debug RIP protocol issues. To trace RIP protocol traffic include the traceoptions statement at the [edit protocols rip] hierarchy level:

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

You can specify the following RIP protocol-specific trace options using the flag statement:

- auth—RIP authentication
- error—RIP error packets
- expiration—RIP route expiration processing
- holddown—RIP hold-down processing
- nsr-synchronization—Nonstop active routing synchronization events
- packets—All RIP packets
- request—RIP information packets
- trigger—RIP triggered updates
- update—RIP update packets

You can optionally specify one or more of the following flag modifiers:

- detail—Detailed trace information
- receive—Packets being received
- send—Packets being transmitted

NOTE: Use the detail flag modifier with caution as this may cause the CPU to become very busy.

Global tracing options are inherited from the configuration set by the traceoptions statement at the [edit routing-options] hierarchy level. You can override the following global trace options for the RIP protocol using the traceoptions flag statement included at the [edit protocols rip] hierarchy level:

- all—All tracing operations
- general—All normal operations and routing table changes (a combination of the normal and route trace operations)
- normal—Normal events
- policy—Policy processing
- route—Routing information
- state—State transitions
- task—Routing protocol task processing
- timer—Routing protocol timer processing

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

Example: Tracing RIP Protocol Traffic

Trace only unusual or abnormal operations to /var/log/routing-log, and trace detailed information about all RIP packets to /var/log/rip-log:

```
[edit]
routing-options {
   traceoptions {
      file /var/log/routing-log;
      flag errors;
   }
}
protocols {
   rip {
      traceoptions {
         file /var/log/rip-log;
         flag packets detail;
      }
   }
}
```

Related Topics

- traceoptions statement
- For general information about tracing and global tracing options, see Tracing Global Routing Protocol Operations on page 129.

Example: Configuring RIP

Configure RIP (for routing instances, include the statements at the [edit routing-instances *routing-instance-name* protocols rip] hierarchy level):

```
[edit policy-options]
policy-statement redist-direct {
   from protocol direct;
   then accept;
}
[edit]
interfaces {
   so-0/0/0 {
      unit 0 {
         family inet;
      }
   }
   at-1/1/0 {
      unit 0 {
         family inet;
      }
   }
   at-1/1/0 {
      unit 42 {
         family inet;
      }
   }
   at-1/1/1 {
      unit 42 {
         family inet;
      }
   }
}
policy-statement redist-direct {
   from protocol direct;
   then accept;
}
[edit protocols rip]
metric-in 3;
receive both;
group wan {
   metric-out 2;
   export redist-direct;
   neighbor so-0/0/0.0;
   neighbor at-1/1/0.0;
   neighbor at-1/1/0.42;
   neighbor at-1/1/1.42 {
      receive version-2;
   }
}
group local {
   neighbor ge-2/3/0.0 {
      metric-in 1;
      send broadcast;
```

```
    }
}
```

Chapter 22

Summary of RIP Configuration Statements

The following sections explain each of the individual RIP statements in the [edit protocols rip] hierarchy. The statements are organized alphabetically.

any-sender

Syntax any-sender;

Hierarchy Level [edit logical-systems *logical-system-name* protocols rip group *group-name* neighbor *neighbor-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols rip group *group-name* neighbor *neighbor-name*],
[edit protocols rip group *group-name* neighbor *neighbor-name*],
[edit routing-instances *routing-instance-name* protocols rip group *group-name* neighbor *neighbor-name*]

Release Information Statement introduced in JUNOS Release 8.0.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Disable strict sender address checks.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Disabling Strict Address Checking for RIP Messages on page 600

authentication-key

Syntax authentication-key *password*;

Hierarchy Level
[edit logical-systems *logical-system-name* protocols rip],
[edit logical-systems *logical-system-name* protocols rip group *group-name* neighbor *neighbor-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols rip],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols rip group *group-name* neighbor *neighbor-name*],
[edit protocols rip],
[edit protocols rip group *group-name* neighbor *neighbor-name*],
[edit routing-instances *routing-instance-name* protocols rip],
[edit routing-instances *routing-instance-name* protocols rip group *group-name* neighbor *neighbor-name*]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Require authentication for RIP route queries received on an interface.

Options *password*—Authentication password. If the password does not match, the packet is rejected. The password can be from 1 through 16 contiguous characters long and can include any ASCII strings.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring Authentication for RIP on page 587

authentication-type

Syntax authentication-type *type*;

Hierarchy Level [edit logical-systems *logical-system-name* protocols rip],
[edit logical-systems *logical-system-name* protocols rip group *group-name* neighbor *neighbor-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols rip],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols rip group *group-name* neighbor *neighbor-name*],
[edit protocols rip],
[edit protocols rip group *group-name* neighbor *neighbor-name*],
[edit routing-instances *routing-instance-name* protocols rip],
[edit routing-instances *routing-instance-name* protocols rip group *group-name* neighbor *neighbor-name*]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Configure the type of authentication for RIP route queries received on an interface.

Default If you do not include this statement and the authentication-key statement, RIP authentication is disabled.

Options *type*—Authentication type:

- md5—Use the MD5 algorithm to create an encoded checksum of the packet. The encoded checksum is included in the transmitted packet. The receiving routing device uses the authentication key to verify the packet, discarding it if the digest does not match. This algorithm provides a more secure authentication scheme.
- none—Disable authentication. If none is configured, the configured authentication key is ignored.
- simple—Use a simple password. The password is included in the transmitted packet, which makes this method of authentication relatively insecure. The password can be from 1 through 16 contiguous letters or digits long.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- authentication-key
- Configuring Authentication for RIP on page 587

bfd-liveness-detection

Syntax

```
bfd-liveness-detection {
    authentication {
        algorithm algorithm-name;
        key-chain key-chain-name;
        <loose-check>;
    }
    detection-time {
        threshold milliseconds;
    }
    minimum-interval milliseconds;
    minimum-receive-interval milliseconds;
    transmit-interval {
        threshold milliseconds;
        minimum-interval milliseconds;
    }
    multiplier number;
    no-adaptation;
    version (1 | automatic);
}
```

Hierarchy Level

[edit logical-systems *logical-system-name* protocols rip group *group-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols rip group *group-name* neighbor *neighbor-name*]
[edit protocols rip group *group-name*],
[edit routing-instances *routing-instance-name* protocols rip group *group-name* neighbor *neighbor-name*]

Release Information

Statement introduced in JUNOS Release 8.0.
Statement introduced in JUNOS Release 9.0 for EX Series switches.
detection-time threshold and transmit-interval threshold options introduced in JUNOS Release 8.2.
Support for logical systems introduced in JUNOS Release 8.3.
no-adaptation statement introduced in JUNOS Release 9.0.
authentication algorithm, authentication key-chain, and authentication loose-check statements introduced in JUNOS Release 9.6.

Description

Configure bidirectional failure detection timers and authentication.

Options

authentication algorithm *algorithm-name* —Configure the algorithm used to authenticate the specified BFD session: simple-password, keyed-md5, keyed-sha-1, meticulous-keyed-md5, or meticulous-keyed-sha-1.

authentication key-chain *key-chain-name*—Associate a security key with the specified BFD session using the name of the security keychain. The name you specify must match one of the keychains configured in the authentication-key-chains key-chain statement at the [edit security] hierarchy level.

authentication loose-check—(Optional) Configure loose authentication checking on the BFD session. Use only for transitional periods when authentication may not be configured at both ends of the BFD session.

detection-time threshold *milliseconds*—Configure a threshold. When the BFD session detection time adapts to a value equal to or greater than the threshold, a single trap and a single system log message are sent.

minimum-interval *milliseconds*—Configure the minimum intervals at which the local routing device transmits a hello packet and then expects to receive a reply from the neighbor with which it has established a BFD session.
Range: 1 through 255,000 milliseconds

minimum-receive-interval *milliseconds*—Configure only the minimum interval at which the local routing device expects to receive a reply from a neighbor with which it has established a BFD session.
Range: 1 through 255,000 milliseconds

multiplier *number*—Configure the number of hello packets not received by a neighbor that causes the originating interface to be declared down.
Range: 1 through 255
Default: 3

no-adaptation—Configure BFD sessions not to adapt to changing network conditions. We recommend that you not disable BFD adaptation unless it is preferable not to have BFD adaptation enabled in your network.

transmit-interval threshold *milliseconds*—Configure a threshold. When the BFD session transmit interval adapts to a value greater than the threshold, a single trap and a single system log message are sent. The interval threshold must be greater than the minimum transmit interval.
Range: 0 through 4,294,967,295 (2^{32} – 1)

transmit-interval minimum-interval *milliseconds*—Configure only a minimum interval at which the local routing device transmits hello packets to a neighbor.
Range: 1 through 255,000

version—Specify the BFD version to detect.
Range: (BFD version 1), or automatic (autodetect the version)
Default: automatic

The remaining statements are explained separately.

Required Privilege Level

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

Related Topics

- Configuring BFD for RIP on page 588
- Configuring BFD Authentication for RIP on page 592

check-zero

Syntax (check-zero | no-check-zero);

Hierarchy Level
[edit logical-systems *logical-system-name* protocols rip],
[edit logical-systems *logical-system-name* protocols rip group *group-name* neighbor *neighbor-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols rip],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols rip group *group-name* neighbor *neighbor-name*],
[edit protocols rip],
[edit protocols rip group *group-name* neighbor *neighbor-name*],
[edit routing-instances *routing-instance-name* protocols rip],
[edit routing-instances *routing-instance-name* protocols rip group *group-name* neighbor *neighbor-name*]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Check whether the reserved fields in a RIP packet are zero:

- check-zero—Discard version 1 packets that have nonzero values in the reserved fields and version 2 packets that have nonzero values in the fields that must be zero. This default behavior implements the RIP version 1 and version 2 specifications.
- no-check-zero—Receive RIP version 1 packets with nonzero values in the reserved fields or RIP version 2 packets with nonzero values in the fields that must be zero. This is in spite of the fact that they are being sent in violation of the specifications in RFC 1058 and RFC 2453.

Default check-zero

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Accepting RIP Packets with Nonzero Values in Reserved Fields on page 595

export

Syntax export [*policy-names*];

Hierarchy Level [edit logical-systems *logical-system-name* protocols rip group *group-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols rip group *group-name*],
[edit protocols rip group *group-name*],
[edit routing-instances *routing-instance-name* protocols rip group *group-name*]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Apply a policy to routes being exported to the neighbors.

Options *policy-names*—Name of one or more policies.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- import
- Applying Policies to Routes Exported by RIP on page 599
- *JUNOS Policy Framework Configuration Guide*

graceful-restart

Syntax

```
graceful-restart {
    disable;
    restart-time seconds;
}
```

Hierarchy Level [edit logical-systems *logical-system-name* protocols rip],
[edit protocols rip]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Configure graceful restart for RIP.

Options disable—Disables graceful restart for RIP.

seconds—Estimated time for the restart to finish, in seconds.
Range: 1 through 600 seconds
Default: 60 seconds

The remaining statements are explained separately.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics

- Configuring Graceful Restart on page 124
- Configuring Graceful Restart for RIP on page 600

group

Syntax

```
group group-name {
    bfd-liveness-detection {
        authentication {
            algorithm algorithm-name;
            key-chain key-chain-name;
            loose-check;
        }
        detection-time {
            threshold milliseconds;
        }
        minimum-interval milliseconds;
        minimum-receive-interval milliseconds;
        transmit-interval {
            threshold milliseconds;
            minimum-interval milliseconds;
        }
        multiplier number;
        version (0 | 1 | automatic);
    }
    preference number;
    metric-out metric;
    export policy;
    route-timeout seconds;
    update-interval seconds;
    neighbor neighbor-name {
        authentication-key password;
        authentication-type type;
        bfd-liveness-detection {
            authentication {
                algorithm algorithm-name;
                key-chain key-chain-name;
                loose-check;
            }
            detection-time {
                threshold milliseconds;
            }
            minimum-interval milliseconds;
            minimum-receive-interval milliseconds;
            transmit-interval {
                threshold milliseconds;
                minimum-interval milliseconds;
            }
            multiplier number;
            version (0 | 1 | automatic);
        }
        (check-zero | no-check-zero);
        import policy-name;
        message-size number;
        metric-in metric;
        metric-out metric;
        receive receive-options;
```

```
        route-timeout seconds;
        send send-options;
        update-interval seconds;
    }
}
```

Hierarchy Level

[edit logical-systems *logical-system-name* protocols rip],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols rip],
[edit protocols rip],
[edit routing-instances *routing-instance-name* protocols rip]

Release Information

Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description

Configure a set of RIP neighbors that share an export policy and metric. The export policy and metric govern what routes to advertise to neighbors in a given group.

Options

group-name—Name of a group, up to 16 characters long.

The remaining statements are explained separately.

Required Privilege Level

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

Related Topics ■ Configuring Group-Specific RIP Properties on page 598

holddown

Syntax holddown *seconds*;

Hierarchy Level [edit logical-systems *logical-system-name* protocols rip],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols rip],
[edit protocols rip],
[edit routing-instances *routing-instance-name* protocols rip]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Configure the time period the expired route is retained in the routing table before being removed.

Options *seconds*—Estimated time to wait before making updates to the routing table.
Range: 10 through 180 seconds
Default: 180 seconds

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics ■ Configuring RIP Timers on page 597

import

Syntax import [*policy-names*];

Hierarchy Level [edit logical-systems *logical-system-name* protocols rip],
[edit logical-systems *logical-system-name* protocols rip group *group-name* neighbor *neighbor-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols rip],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols rip group *group-name* neighbor *neighbor-name*],
[edit protocols rip],
[edit protocols rip group *group-name* neighbor *neighbor-name*],
[edit routing-instances *routing-instance-name* protocols rip],
[edit routing-instances *routing-instance-name* protocols rip group *group-name* neighbor *neighbor-name*]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Apply one or more policies to routes being imported by the local router from its neighbors.

Options *policy-names*—Name of one or more policies.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- export
- Applying Policies to RIP Routes Imported from Neighbors on page 596
- *JUNOS Policy Framework Configuration Guide*

message-size

Syntax message-size *number*;

Hierarchy Level
[edit logical-systems *logical-system-name* protocols rip],
[edit logical-systems *logical-system-name* protocols rip group *group-name* neighbor *neighbor-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols rip],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols rip group *group-name* neighbor *neighbor-name*],
[edit protocols rip],
[edit protocols rip group *group-name* neighbor *neighbor-name*],
[edit routing-instances *routing-instance-name* protocols rip],
[edit routing-instances *routing-instance-name* protocols rip group *group-name* neighbor *neighbor-name*]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Specify the number of route entries to be included in every RIP update message. To ensure interoperability with other vendors' equipment, use the standard of 25 route entries per message.

Options *number*—Number of route entries per update message.
Range: 25 through 255 entries
Default: 25 entries

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring the Number of Route Entries in RIP Update Messages on page 596

metric-in

Syntax metric-in *metric*;

Hierarchy Level
[edit logical-systems *logical-system-name* protocols rip],
[edit logical-systems *logical-system-name* protocols rip group *group-name* neighbor *neighbor-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols rip],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols rip group *group-name* neighbor *neighbor-name*],
[edit protocols rip],
[edit protocols rip group *group-name* neighbor *neighbor-name*],
[edit routing-instances *routing-instance-name* protocols rip],
[edit routing-instances *routing-instance-name* protocols rip group *group-name* neighbor *neighbor-name*]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Specify the metric to add to incoming routes when advertising into RIP routes that were learned from other protocols. Use this statement to configure the routing device to prefer RIP routes learned through a specific neighbor.

Options *metric*—Metric value.
Range: 1 through 16
Default: 1

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring the Metric Value Added to Imported RIP Routes on page 596

metric-out

Syntax metric-out *metric*;

Hierarchy Level [edit logical-systems *logical-system-name* protocols rip group *group-name* neighbor *neighbor-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols rip group *group-name* neighbor *neighbor-name*],
[edit protocols rip group *group-name* neighbor *neighbor-name*],
[edit routing-instances *routing-instance-name* protocols rip group *group-name* neighbor *neighbor-name*]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Specify the metric value to add to routes transmitted to the neighbor. Use this statement to control how other routing devices prefer RIP routes sent from this neighbor.

Options *metric*—Metric value.
Range: 1 through 16
Default: 1

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring the Metric for Routes Exported by RIP on page 600

neighbor

Syntax

```
neighbor neighbor-name {
    authentication-key password;
    authentication-type type;
    bfd-liveness-detection {
        authentication {
            algorithm algorithm-name;
            key-chain key-chain-name;
            loose-check;
        }
        detection-time {
            threshold milliseconds;
        }
        minimum-interval milliseconds;
        minimum-receive-interval milliseconds;
        transmit-interval {
            threshold milliseconds;
            minimum-interval milliseconds;
        }
        multiplier number;
        version (0 | 1 | automatic);
    }
    (check-zero | no-check-zero);
    import policy-name;
    message-size number;
    metric-in metric;
    metric-out metric;
    receive receive-options;
    route-timeout seconds;
    send send-options;
    update-interval seconds;
}
```

Hierarchy Level

[edit logical-systems *logical-system-name* protocols rip group *group-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols rip group *group-name*],
[edit protocols rip group *group-name*],
[edit routing-instances *routing-instance-name* protocols rip group *group-name*]

Release Information

Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description

Configure neighbor-specific RIP parameters, thereby overriding the defaults set for the routing device.

Options

neighbor-name—Name of an interface over which a routing device communicates to its neighbors.

The remaining statements are explained separately.

Required Privilege Level

routing—To view this statement in the configuration.

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

Related Topics ■ Overview of RIP Neighbor Properties on page 586

no-check-zero

See check-zero

preference

Syntax preference *preference*;

Hierarchy Level [edit logical-systems *logical-system-name* protocols rip group *group-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols rip group *group-name*],
[edit protocols rip group *group-name*],
[edit routing-instances *routing-instance-name* protocols rip group *group-name*]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Specify the preference of external routes learned by RIP as compared to those learned from other routing protocols.

Options *preference*—Preference value. A lower value indicates a more preferred route.
Range: 0 through 4,294,967,295 ($2^{32} - 1$)
Default: 100

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring the Default Preference Value for RIP on page 599

receive

Syntax receive *receive-options*;

Hierarchy Level [edit logical-systems *logical-system-name* protocols rip],
[edit logical-systems *logical-system-name* protocols rip group *group-name* neighbor *neighbor-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols rip],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols rip group *group-name* neighbor *neighbor-name*],
[edit protocols rip],
[edit protocols rip group *group-name* neighbor *neighbor-name*],
[edit routing-instances *routing-instance-name* protocols rip],
[edit routing-instances *routing-instance-name* protocols rip group *group-name* neighbor *neighbor-name*]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Configure RIP receive options.

Options *receive-options*—One of the following:

- both—Accept both RIP version 1 and version 2 packets.
- none—Do not receive RIP packets.
- version-1—Accept only RIP version 1 packets.
- version-2—Accept only RIP version 2 packets.

  Default: both

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics

- send
- Configuring RIP Update Messages on page 597

rib-group

Syntax rib-group *group-name*;

Hierarchy Level [edit logical-systems *logical-system-name* protocols rip],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols rip],
[edit protocols rip],
[edit routing-instances *routing-instance-name* protocols rip]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Install RIP routes into multiple routing tables by configuring a routing table group.

Options *group-name*—Name of the routing table group.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring Routing Table Groups for RIP on page 597

rip

Syntax rip {...}

Hierarchy Level [edit logical-systems *logical-system-name* protocols],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols],
[edit protocols],
[edit routing-instances *routing-instance-name* protocols]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Enable RIP routing on the routing device.

Default RIP is disabled on the routing device.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Minimum RIP Configuration on page 585

route-timeout

Syntax route-timeout *seconds*;

Hierarchy Level [edit logical-systems *logical-system-name* protocols rip],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols rip],
[edit logical-systems *logical-system-name* protocols rip group *group-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols rip group *group-name*],
[edit protocols rip],
[edit protocols rip group *group-name*],
[edit routing-instances *routing-instance-name* protocols rip],
[edit routing-instances *routing-instance-name* protocols rip group *group-name*]

Release Information Statement introduced in JUNOS Release 7.6.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Configure the route timeout interval for RIP.

Options *seconds*—Estimated time to wait before making updates to the routing table.
Range: 30 through 360 seconds
Default: 180 seconds

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring RIP Timers on page 597

send

Syntax send *send-options*;

Hierarchy Level [edit logical-systems *logical-system-name* protocols rip],
[edit logical-systems *logical-system-name* protocols rip group *group-name* neighbor *neighbor-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols rip],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols rip group *group-name* neighbor *neighbor-name*],
[edit protocols rip],
[edit protocols rip group *group-name* neighbor *neighbor-name*],
[edit routing-instances *routing-instance-name* protocols rip],
[edit routing-instances *routing-instance-name* protocols rip group *group-name* neighbor *neighbor-name*]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Configure RIP send options.

Options *send-options*—One of the following:

- broadcast—Broadcast RIP version 2 packets (RIP version 1 compatible).
- multicast—Multicast RIP version 2 packets. This is the default.
- none—Do not send RIP updates.
- version-1—Broadcast RIP version 1 packets.

Default: multicast

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- receive
- Configuring RIP Update Messages on page 597

traceoptions

Syntax

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

Hierarchy Level

[edit logical-systems *logical-system-name* protocols rip],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols rip],
[edit protocols rip],
[edit routing-instances *routing-instance-name* protocols rip]

Release Information

Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description

Set RIP protocol-level tracing options.

Default

The default RIP protocol-level trace options are inherited from the global **traceoptions** statement.

Options

disable—(Optional) Disable the tracing operation. One use of this option is to disable a single operation when you have defined a broad group of tracing operations, such as **all**.

file *filename*—Name of the file to receive the output of the tracing operation. Enclose the name in quotation marks. We recommend that you place RIP tracing output in the file **/var/log/rip-log**.

files *number*—(Optional) Maximum number of trace files. When a trace file named *trace-file* reaches its maximum size, it is renamed *trace-file*.0, then *trace-file*.1, and so on, until the maximum number of trace files is reached. Then, the oldest trace file is overwritten.
If you specify a maximum number of files, you must also specify a maximum file size with the size option.
Range: 2 through 1000 files
Default: 10 files

flag—Tracing operation to perform. To specify more than one tracing operation, include multiple **flag** statements.

RIP Tracing Options

- **auth**—RIP authentication
- **error**—RIP error packets
- **expiration**—RIP route expiration processing
- **holddown**—RIP hold-down processing
- **nsr-synchronization**—Nonstop routing synchronization events
- **packets**—All RIP packets

- request—RIP information packets such as request, poll, and poll entry packets
- trigger—RIP triggered updates
- update—RIP update packets

Global Tracing Options

- all—All tracing operations
- general—A combination of the normal and route trace operations
- normal—All normal operations

  Default: If you do not specify this option, only unusual or abnormal operations are traced.
- policy—Policy operations and actions
- route—Routing table changes
- state—State transitions
- task—Routing protocol task processing
- timer—Routing protocol timer processing

flag-modifier—(Optional) Modifier for the tracing flag. You can specify one or more of these modifiers:

- detail—Provide detailed trace information
- receive—Packets being received
- receive-detail—Provide detailed trace information for packets being received
- send—Packets being transmitted
- send-detail—Provide detailed trace information for packets being transmitted

no-world-readable—(Optional) Prevent any user from reading the log file.

size *size*—(Optional) Maximum size of each trace file, in kilobytes (KB) or megabytes (MB). When a trace file named *trace-file* reaches this size, it is renamed *trace-file*.0. When the *trace-file* again reaches its maximum size, *trace-file*.0 is renamed *trace-file*.1 and *trace-file* is renamed *trace-file*.0. This renaming scheme continues until the maximum number of trace files is reached. Then, the oldest trace file is overwritten.
If you specify a maximum file size, you must also specify a maximum number of trace files with the files option.
Syntax: *x*k to specify KB, *x*m to specify MB, or *x*g to specify GB
Range: 10 KB through the maximum file size supported on your system
Default: 128 KB

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

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics ■ Tracing RIP Protocol Traffic on page 601

update-interval

Syntax update-interval *seconds*;

Hierarchy Level [edit logical-systems *logical-system-name* protocols rip],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols rip],
[edit protocols rip],
[edit routing-instances *routing-instance-name* protocols rip]

Release Information Statement introduced in JUNOS Release 7.6.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Configure an update time interval to periodically send out routes learned by RIP to neighbors.

Options *seconds*—Estimated time to wait before making updates to the routing table.
Range: 10 through 60 seconds
Default: 30 seconds

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics ■ Configuring RIP Timers on page 597

Chapter 23

Introduction to RIPng

This chapter discusses the following topics that provide background information about RIP next generation (RIPng):

- RIPng Overview on page 631
- RIPng Standards on page 632

RIPng Overview

RIP next generation (RIPng) is an interior gateway protocol (IGP) that uses a distance-vector algorithm to determine the best route to a destination, using the hop count as the metric. RIPng is a routing protocol that exchanges routing information used to compute routes and is intended for IP version 6 (IPv6)-based networks.

This section discusses the following topics:

- RIPng Protocol Overview on page 631
- RIPng Packets on page 632

RIPng Protocol Overview

The RIPng IGP uses the Bellman-Ford *distance-vector* algorithm to determine the best route to a destination. RIPng uses the hop count as the metric. RIPng allows hosts and routers to exchange information for computing routes through an IP-based network. RIPng is intended to act as an IGP for moderately-sized autonomous systems (ASs).

The JUNOS Software implementation of RIPng is similar to RIPv2. However, RIPng is a distinct routing protocol from RIPv2 and has the following differences:

- RIPng does not need to implement authentication on packets.
- There is no support for multiple instances of RIPng.
- There is no support for RIPng routing table groups.

RIPng is a User Datagram Protocol (UDP)-based protocol and uses UDP port 521.

RIPng has the following architectural limitations:

- The longest network path cannot exceed 15 hops (assuming that each network, or hop, has a cost of 1).

- RIPng depends on counting to infinity to resolve certain unusual situations. When the network consists of several hundred routers, and when a routing loop has formed, the amount of time and network bandwidth required to resolve a next hop might be great.
- RIPng uses only a fixed metric to select a route. Other IGPs use additional parameters, such as measured delay, reliability, and load.

RIPng Packets

A RIPng packet header contains the following fields:

- Command—Indicates whether the packet is a request or response message. Request messages seek information for the router's routing table. Response messages are sent periodically or when a request message is received. Periodic response messages are called *update messages*. Update messages contain the command and version fields and a set of destinations and metrics.
- Version number—Specifies the version of RIPng that the originating router is running. This is currently set to Version 1.

The rest of the RIPng packet contains a list of routing table entries that contain the following fields:

- Destination prefix—128-bit IPv6 address prefix for the destination.
- Prefix length—Number of significant bits in the prefix.
- Metric—Value of the metric advertised for the address.
- Route tag—A route attribute that must be advertised and redistributed with the route. Primarily, the route tag distinguishes external RIPng routes from internal RIPng routes in cases where routes must be redistributed across an exterior gateway protocol (EGP).

RIPng Standards

RIPng is defined in the following documents:

- RFC 2080, *RIPng for IPv6*
- RFC 2081, *RIPng Protocol Applicability Statement*

To access Internet Requests for Comments (RFCs) and drafts, go to the Internet Engineering Task Force (IETF) Web site at http://www.ietf.org.

Chapter 24

RIPng Configuration Guidelines

This chapter discusses the following topics that provide information for configuring and monitoring RIPng:

- Configuring RIPng on page 633
- Minimum RIPng Configuration on page 634
- Overview of RIPng Global Properties on page 635
- Overview of RIPng Neighbor Properties on page 635
- Applying Policies to RIPng Routes Imported from Neighbors on page 636
- Configuring the Metric Value Added to Imported RIPng Routes on page 636
- Configuring RIPng Update Messages on page 636
- Configuring RIPng Timers on page 637
- Configuring Group-Specific RIPng Properties on page 637
- Configuring Graceful Restart for RIPng on page 639
- Tracing RIPng Protocol Traffic on page 639
- Example: Configuring RIPng on page 641

Configuring RIPng

To configure RIP next generation (RIPng), you include the following statements:

```
protocols {
    ripng {
        graceful-restart {
            disable;
            restart-time seconds;
        }
        holddown seconds;
        import [ policy-names ];
        metric-in metric;
        receive <none>;
        route-timeout seconds;
        send <none>;
        update-interval seconds;
        traceoptions {
            file filename <files number> <size size> <world-readable | no-world-readable>;
            flag flag <flag-modifier> <disable>;
        }
```

```
        group group-name {
            export [ policy-names ];
            metric-out metric;
            preference number;
            route-timeout seconds;
            update-interval seconds;
            neighbor neighbor-name {
                import [ policy-names ];
                metric-in metric;
                receive <none>;
                route-timeout seconds;
                send <none>;
                update-interval seconds;
            }
        }
    }
}
```

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for these statements.

By default, RIPng is disabled.

NOTE: By default, RIPng routes are not redistributed. You must configure export policy needs to redistribute RIPng routes.

To have a router exchange routes with other routers, you must configure RIPng groups and neighbors. RIPng routes received from routers not configured as RIPng neighbors are ignored. Likewise, RIPng routes are advertised only to routers configured as RIPng neighbors.

For a configuration example, see “Example: Configuring RIPng” on page 641.

Minimum RIPng Configuration

For a routing device to accept RIPng routes, you must configure at least one RIPng group and the associated neighbor. Routes received from routing devices that are not configured as neighbors are ignored. All other RIPng configuration statements are optional. Include one neighbor statement for each interface on which you want to receive routes. The local routing device imports all routes by default from this neighbor and does not advertise routes.

```
[edit]
protocols {
    ripng {
        group group-name {
            neighbor interface-name;
        }
    }
}
```

NOTE: When you configure RIPng on an interface, you must also include the family inet statement at the [edit interfaces *interface-name* unit *logical-unit-number*] hierarchy level.

Overview of RIPng Global Properties

To define RIPng global properties, which apply to all RIPng neighbors, include one or more of the following statements.

```
import [ policy-names ];
metric-in metric;
receive receive-options;
send send-options;
```

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for these statements.

For more information about configuring RIPng global properties, see the following topics:

- Applying Policies to RIPng Routes Imported from Neighbors on page 636
- Configuring the Metric Value Added to Imported RIPng Routes on page 636
- Configuring RIPng Update Messages on page 636

Overview of RIPng Neighbor Properties

To define neighbor-specific properties, include one or more of the following statements.

```
neighbor neighbor-name {
    import [ policy-names ];
    metric-in metric;
    receive receive-options;
    send send-options;
}
```

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for these statements.

For more information about configuring RIPng neighbor properties, see the following topics:

- Applying Policies to RIPng Routes Imported from Neighbors on page 636
- Configuring the Metric Value Added to Imported RIPng Routes on page 636
- Configuring RIPng Update Messages on page 636

Applying Policies to RIPng Routes Imported from Neighbors

To filter routes being imported by the local routing device from its neighbors, include the import statement and list the names of one or more policies to be evaluated. If you specify more than one policy, they are evaluated in order (first to last) and the first matching policy is applied to the route. If no match is found, the local routing device does not import any routes.

```
import [ policy-names ];
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

Configuring the Metric Value Added to Imported RIPng Routes

By default, RIPng imports routes from the neighbors configured with the neighbor statement. These routes include those learned from RIPng as well as those learned from other protocols. By default, the current route metric of routes that RIPng imports from its neighbors is incremented by 1.

To change the default metric to be added to incoming routes, include the metric-in statement:

```
metric-in metric;
```

metric can be a value from 1 through 15. A value of 16 indicates infinity, or unreachable.

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

Configuring RIPng Update Messages

You can enable and disable the sending or receiving of update messages. By default, sending and receiving update messages is enabled. To disable the sending and receiving of update messages, include the receive none and send none statements:

```
receive none;
send none;
```

To enable the sending and receiving of update messages, include the receive and send statements:

```
receive;
send;
```

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for these statements.

Configuring RIPng Timers

You can configure various timers for RIPng.

RIPng routes expire when either a route timeout limit is met or a route metric reaches infinity, and the route is no longer valid. However, the expired route is retained in the routing table for a time period so that neighbors can be notified that the route has been dropped. This time period is set by configuring the hold-down timer. Upon expiration of the hold-down timer, the route is removed from the routing table.

To configure the hold-down timer for RIPng, include the holddown statement:

```
holddown seconds;
```

seconds can be a value from 10 through 180. The default value is 120 seconds.

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

You can set a route timeout interval. If a route is not refreshed after being installed into the routing table by the specified time interval, the route is removed from the routing table.

To configure the route timeout for RIPng, include the route-timeout statement:

```
route-timeout seconds;
```

seconds can be a value from 30 through 360. The default value is 180 seconds.

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

You can set an update time interval to periodically send out routes learned by RIPng to neighbors.

To configure the update time interval, include the update-interval statement:

```
update-interval seconds;
```

seconds can be a value from 10 through 60. The default value is 30 seconds.

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

Configuring Group-Specific RIPng Properties

You can group together neighbors that share the same export policy and export metric defaults. You configure group-specific RIPng properties by including the group statement:

```
group group-name {
    export [ policy-names ];
```

```
    metric-out metric;
    neighbor {
        ... neighbor-options ...
    }
    preference number;
}
```

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for these statements.

Each group must contain at least one neighbor. You should create a group for each export policy that you have. For information about configuring neighbors, see "Overview of RIPng Neighbor Properties" on page 635.

This section discusses the following tasks:

- Applying Policies to Routes Exported by RIPng on page 638
- Configuring the Default Preference Value for RIPng on page 638
- Configuring the Metric for Routes Exported by RIPng on page 639

Applying Policies to Routes Exported by RIPng

By default, RIPng does not export routes it has learned to its neighbors. To have RIPng export routes, apply one or more export policies. To apply export policies and to filter routes being exported from the local routing device to its neighbors, include the export statement and list the name of the policy to be evaluated:

```
export [ policy–names ];
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

You can define one or more export policies. If no routes match the policies, the local routing device does not export any routes to its neighbors. Export policies override any metric values determined through calculations involving the values configured with the metric-in and metric-out statements (discussed in "Configuring the Metric Value Added to Imported RIPng Routes" on page 636 and "Configuring the Metric for Routes Exported by RIPng" on page 639 respectively).

Configuring the Default Preference Value for RIPng

By default, the JUNOS Software assigns a preference of 100 to routes that originate from RIPng. When the JUNOS Software determines that a route preference is to become the active route, the software selects the route with the lowest preference and installs this route into the forwarding table.

To modify the default RIPng preference value, include the preference statement:

```
preference preference;
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

preference can be a value from 0 through 4,294,967,295 ($2^{32} - 1$).

Configuring the Metric for Routes Exported by RIPng

If you configure an export policy, RIPng exports routes it has learned to the neighbors configured with the neighbor statement.

If a route being exported was learned from a member of the same RIPng group, the metric associated with that route (unless modified by an export policy) is the normal RIPng metric. For example, a RIPng route with a metric of 5 learned from a neighbor configured with a metric-in value of 2 is advertised with a combined metric of 7 when advertised to RIPng neighbors in the same group. However, if this route was learned from a RIPng neighbor in a different group or from a different protocol, the route is advertised with the metric value configured for that group with the metric-out statement. The default value for metric-out is 1.

To modify the metric for routes advertised outside a group, include the metric-out statement:

```
metric-out metric;
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

Configuring Graceful Restart for RIPng

Graceful restart is disabled by default. You can globally enable graceful restart for all routing protocols under the [edit routing-options] hierarchy level.

You can configure graceful restart parameters specifically for RIPng. To do this, include the graceful-restart statement:

```
graceful-restart {
   restart-time seconds;
}
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

To disable graceful restart for RIPng, specify the disable statement. To configure a time period for the restart to finish, specify the restart-time statement.

Tracing RIPng Protocol Traffic

You can trace various RIPng protocol traffic to help debug RIP protocol issues. To trace RIP protocol traffic include the traceoptions statement at the [edit protocols ripng] hierarchy level:

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

You can specify the following RIPng protocol-specific trace options using the flag statement:

- error—RIPng error packets
- expiration—RIPng route expiration processing
- holddown—RIPng hold-down processing
- nsr-synchronization—Nonstop routing synchronization events
- packets—All RIPng packets
- request—RIPng information packets
- trigger—RIPng triggered updates
- update—RIPng update packets

You can optionally specify one or more of the following flag modifiers:

- detail—Detailed trace information
- receive—Packets being received
- send—Packets being transmitted

NOTE: Use the detail flag modifier with caution as this may cause the CPU to become very busy.

Global tracing options are inherited from the configuration set by the traceoptions statement at the [edit routing-options] hierarchy level. You can override the following global trace options for the RIPng protocol using the traceoptions flag statement included at the [edit protocols ripng] hierarchy level:

- all—All tracing operations
- general—All normal operations and routing table changes (a combination of the normal and route trace operations)
- normal—Normal events
- policy—Policy processing
- route—Routing information
- state—State transitions
- task—Routing protocol task processing
- timer—Routing protocol timer processing

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

Related Topics
- traceoptions statement

- For general information about tracing and global tracing options, see Tracing Global Routing Protocol Operations on page 129.

Example: Configuring RIPng

Configure RIPng:

```
[edit policy-options]
policy-statement redist-direct {
    from protocol direct;
    then accept;
}
[edit protocols ripng]
metric-in 3;
group wan {
    metric-out 2;
    export redist-direct;
    neighbor so-0/0/0.0;
    neighbor at-1/1/0.0;
    neighbor at-1/1/0.42;
    neighbor at-1/1/1.42 {
        receive version-2;
    }
}
group local {
    neighbor ge-2/3/0.0 {
        metric-in 1;
        send broadcast;
    }
}
```

Chapter 25

Summary of RIPng Configuration Statements

The following sections explain each of the RIP next generation (RIPng) statements in the [edit protocols ripng] hierarchy. The statements are organized alphabetically.

export

Syntax export [*policy-names*];

Hierarchy Level [edit logical-systems *logical-system-name* protocols ripng group *group-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ripng group *group-name*],
[edit protocols ripng group *group-name*],
[edit routing-instances *routing-instance-name* protocols ripng group *group-name*]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.
Support for routing instances introduced in JUNOS Release 9.0.

Description Apply a policy or list of policies to routes being exported to the neighbors.

Options *policy-names*—Name of one or more policies.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- import
- Applying Policies to Routes Exported by RIPng on page 638

graceful-restart

Syntax

```
graceful-restart {
    disable;
    restart-time seconds;
}
```

Hierarchy Level

[edit logical-systems *logical-system-name* protocols ripng],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ripng],
[edit protocols ripng],
[edit routing-instances *routing-instance-name* protocols ripng]

Release Information

Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.
Support for routing instances introduced in JUNOS Release 9.0.

Description

Configure graceful restart for RIPng.

Options

disable—Disables graceful restart for RIPng.

seconds—Estimated time period for the restart to finish.
Range: 1 through 600 seconds
Default: 60 seconds

Required Privilege Level

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

Related Topics

- Configuring Graceful Restart on page 124
- Configuring Graceful Restart for RIPng on page 639

group

Syntax

```
group group-name {
    export [ policy-names ];
    metric-out metric;
    preference number;
    route-timeout seconds;
    update-interval seconds;
    neighbor neighbor-name {
        import policy-name;
        metric-in metric;
        receive <none>;
        route-timeout seconds;
        send <none>;
        update-interval seconds;
    }
}
```

Hierarchy Level

[edit logical-systems *logical-system-name* protocols ripng],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ripng],
[edit protocols ripng],
[edit routing-instances *routing-instance-name* protocols ripng]

Release Information

Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.
Support for routing instances introduced in JUNOS Release 9.0.

Description

Configure a set of RIPng neighbors that share an export policy and metric. The export policy and metric govern what routes to advertise to neighbors in a given group.

Options

group-name—Name of a group, up to 16 characters long.

The remaining statements are explained separately.

Required Privilege Level

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

Related Topics

- Configuring Group-Specific RIPng Properties on page 637

holddown

Syntax holddown *seconds*;

Hierarchy Level [edit logical-systems *logical-system-name* protocols ripng],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ripng],
[edit protocols ripng],
[edit routing-instances *routing-instance-name* protocols ripng]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.
Support for routing instances introduced in JUNOS Release 9.0.

Description Configure the time period the expired route is retained in the routing table before being removed.

Options *seconds*—Estimated time to wait before making updates to the routing table.
Default: 180 seconds
Range: 10 through 180 seconds

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring RIPng Timers on page 637

import

Syntax import [*policy-names*];

Hierarchy Level
[edit logical-systems *logical-system-name* protocols ripng],
[edit logical-systems *logical-system-name* protocols ripng group *group-name* neighbor *neighbor-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ripng],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ripng group *group-name* neighbor *neighbor-name*],
[edit protocols ripng],
[edit protocols ripng group *group-name* neighbor *neighbor-name*],
[edit routing-instances *routing-instance-name* protocols ripng],
[edit routing-instances *routing-instance-name* protocols ripng group *group-name* neighbor *neighbor-name*]

Release Information
Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.
Support for routing instances introduced in JUNOS Release 9.0.

Description Apply one or more policies to routes being imported into the local routing device from the neighbors.

Options *policy-names*—Name of one or more policies.

Required Privilege Level
routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- export
- Applying Policies to RIPng Routes Imported from Neighbors on page 636

metric-in

Syntax metric-in *metric*;

Hierarchy Level
[edit logical-systems *logical-system-name* protocols ripng],
[edit logical-systems *logical-system-name* protocols ripng group *group-name* neighbor *neighbor-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ripng],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ripng group *group-name* neighbor *neighbor-name*],
[edit protocols ripng],
[edit protocols ripng group *group-name* neighbor *neighbor-name*],
[edit routing-instances *routing-instance-name* protocols ripng],
[edit routing-instances *routing-instance-name* protocols ripng group *group-name* neighbor *neighbor-name*]

Release Information
Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.
Support for routing instances introduced in JUNOS Release 9.0.

Description Specify the metric to add to incoming routes when advertising into RIPng routes that were learned from other protocols. Use this statement to configure the routing device to prefer RIPng routes learned through a specific neighbor.

Options
metric—Metric value.
Range: 1 through 16
Default: 1

Required Privilege Level
routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring the Metric Value Added to Imported RIPng Routes on page 636

metric-out

Syntax metric-out *metric*;

Hierarchy Level [edit logical-systems *logical-system-name* protocols ripng group *group-name* neighbor *neighbor-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ripng group *group-name* neighbor *neighbor-name*],
[edit protocols ripng group *group-name* neighbor *neighbor-name*],
[edit routing-instances *routing-instance-name* protocols ripng group *group-name* neighbor *neighbor-name*]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.
Support for routing instances introduced in JUNOS Release 9.0.

Description Specify the metric value to add to routes transmitted to the neighbor. Use this statement to control how other routing devices prefer RIPng routes sent from this neighbor.

Options *metric*—Metric value.
Range: 1 through 16
Default: 1

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring the Metric for Routes Exported by RIPng on page 639

neighbor

Syntax

```
neighbor neighbor-name {
    import [ policy-names ];
    metric-in metric;
    receive <none>;
    route-timeout seconds;
    send <none>;
    update-interval seconds;
}
```

Hierarchy Level

[edit logical-systems *logical-system-name* protocols ripng group *group-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ripng group *group-name*],
[edit protocols ripng group *group-name*],
[edit routing-instances *routing-instance-name* protocols ripng group *group-name*]

Release Information

Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.
Support for routing instances introduced in JUNOS Release 9.0.

Description

Configure neighbor-specific RIPng parameters, thereby overriding the defaults set for the routing device.

Options

neighbor-name—Name of an interface over which a routing device communicates to its neighbors.

The remaining statements are explained separately.

Required Privilege Level

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

Related Topics

- Overview of RIPng Neighbor Properties on page 635

preference

Syntax preference *preference*;

Hierarchy Level [edit logical-systems *logical-system-name* protocols ripng group *group-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ripng group *group-name*],
[edit protocols ripng group *group-name*],
[edit routing-instances *routing-instance-name* protocols ripng group *group-name*]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.
Support for routing instances introduced in JUNOS Release 9.0.

Description Specify the preference of external routes learned by RIPng as compared to those learned from other routing protocols.

Options *preference*—Preference value. A lower value indicates a more preferred route.
Range: 0 through 4,294,967,295 ($2^{32} - 1$)
Default: 100

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring the Default Preference Value for RIPng on page 638

receive

Syntax receive <none>;

Hierarchy Level
[edit logical-systems *logical-system-name* protocols ripng],
[edit logical-systems *logical-system-name* protocols ripng group *group-name* neighbor *neighbor-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ripng],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ripng group *group-name* neighbor *neighbor-name*],
[edit protocols ripng],
[edit protocols ripng group *group-name* neighbor *neighbor-name*],
[edit routing-instances *routing-instance-name* protocols ripng],
[edit routing-instances *routing-instance-name* protocols ripng group *group-name* neighbor *neighbor-name*]

Release Information
Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.
Support for routing instances introduced in JUNOS Release 9.0.

Description Enable or disable receiving of update messages.

Options
none—(Optional) Disable receiving update messages.
Default: Enabled

Required Privilege Level
routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- send
- Configuring RIPng Update Messages on page 636

ripng

Syntax ripng {...}

Hierarchy Level [edit logical-systems *logical-system-name* protocols],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols],
[edit protocols],
[edit routing-instances *routing-instance-name* protocols]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.
Support for routing instances introduced in JUNOS Release 9.0.

Description Enable RIPng routing on the routing device.

Default RIPng is disabled on the routing device.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Minimum RIPng Configuration on page 634

route-timeout

Syntax route-timeout *seconds*;

Hierarchy Level [edit logical-systems *logical-system-name* protocols ripng],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ripng],
[edit protocols ripng],
[edit routing-instances *routing-instance-name* protocols ripng]

Release Information Statement introduced in JUNOS Release 7.6.
Statement introduced in JUNOS Release 9.0 for EX Series switches.
Support for routing instances introduced in JUNOS Release 9.0.

Description Configure the route timeout interval for RIPng.

Options *seconds*—Estimated time to wait before making updates to the routing table.
Range: 30 through 360 seconds
Default: 180 seconds

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring RIPng Timers on page 637

send

Syntax send <none>;

Hierarchy Level [edit logical-systems *logical-system-name* protocols ripng],
[edit logical-systems *logical-system-name* protocols ripng group *group-name* neighbor *neighbor-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instances-name* protocols ripng],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ripng group *group-name* neighbor *neighbor-name*],
[edit protocols ripng],
[edit protocols ripng group *group-name* neighbor *neighbor-name*],
[edit routing-instances *routing-instance-name* protocols ripng],
[edit routing-instances *routing-instance-name* protocols ripng group *group-name* neighbor *neighbor-name*]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.
Support for routing instances introduced in JUNOS Release 9.0.

Description Enable or disable sending of update messages.

Options none—(Optional) Disable sending of update messages.
Default: Enabled

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- receive
- Configuring RIPng Update Messages on page 636

traceoptions

Syntax

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

Hierarchy Level

[edit logical-systems *logical-system-name* protocols ripng],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ripng],
[edit protocols ripng],
[edit routing-instances *routing-instance-name* protocols ripng]

Release Information

Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.
Support for routing instances introduced in JUNOS Release 9.0.

Description

Set RIPng protocol-level tracing options.

Default

The default RIPng protocol-level trace options are inherited from the global traceoptions statement.

Options

disable—(Optional) Disable the tracing operation. One use of this option is to disable a single operation when you have defined a broad group of tracing operations, such as all.

file *filename*—Name of the file to receive the output of the tracing operation. Enclose the name in quotation marks. We recommend that you place RIPng tracing output in the file /var/log/ripng-log.

files *number*—(Optional) Maximum number of trace files. When a trace file named *trace-file* reaches its maximum size, it is renamed *trace-file*.0, then *trace-file*.1, and so on, until the maximum number of trace files is reached. Then, the oldest trace file is overwritten.
If you specify a maximum number of files, you must also specify a maximum file size with the size option.
Range: 2 through 1000 files
Default: 10 files

flag *flag*—Tracing operation to perform. To specify more than one tracing operation, include multiple flag statements.

RIPng Tracing Options

- error—RIPng error packets
- expiration—RIPng route expiration processing
- holddown—RIPng hold-down processing
- nsr-synchronization—Nonstop routing synchronization events
- packets—All RIPng packets

- request—RIPng information packets such as request, poll, and poll entry packets
- trigger—RIPng triggered updates
- update—RIPng update packets

Global Tracing Options

- all—All tracing operations
- general—A combination of the normal and route trace operations
- normal—All normal operations

  Default: If you do not specify this option, only unusual or abnormal operations are traced.
- policy—Policy operations and actions
- route—Routing table changes
- state—State transitions
- task—Routing protocol task processing
- timer—Routing protocol timer processing

flag-modifier—(Optional) Modifier for the tracing flag. You can specify one or more of these modifiers:

- detail—Provide detailed trace information
- receive—Packets being received
- receive-detail—Provide detailed trace information for packets being received
- send—Packets being transmitted
- send-detail—Provide detailed trace information for packets being transmitted

no-world-readable—(Optional) Do not allow any user to read the log file.

size *size*—(Optional) Maximum size of each trace file, in kilobytes (KB), megabytes (MB), or gigabytes (GB). When a trace file named *trace-file* reaches this size, it is renamed *trace-file*.0. When the *trace-file* again reaches its maximum size, *trace-file*.0 is renamed *trace-file*.1 and *trace-file* is renamed *trace-file*.0. This renaming scheme continues until the maximum number of trace files is reached. Then, the oldest trace file is overwritten.
If you specify a maximum file size, you must also specify a maximum number of trace files with the files option.
Syntax: *x*k to specify KB, *x*m to specify MB, or *x*g to specify GB
Range: 10 KB through the maximum file size supported on your system
Default: 128 KB

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

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics ■ Tracing RIPng Protocol Traffic on page 639

update-interval

Syntax update-interval *seconds*;

Hierarchy Level [edit logical-systems *logical-system-name* protocols ripng],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols ripng],
[edit protocols ripng],
[edit routing-instances *routing-instance-name* protocols ripng]

Release Information Statement introduced in JUNOS Release 7.6.
Statement introduced in JUNOS Release 9.0 for EX Series switches.
Support for routing instances introduced in JUNOS Release 9.0.

Description Configure an update time interval to periodically send out routes learned by RIP to neighbors.

Options *seconds*—Estimated time to wait before making updates to the routing table.
Range: 10 through 60 seconds
Default: 30 seconds

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics ■ Configuring RIP Timers on page 597

Chapter 26

Introduction to ICMP Router Discovery

This chapter discusses the following topics that provide background information about ICMP router discovery:

- ICMP Router Discovery Overview on page 659
- ICMP Router Discovery Standards on page 660

ICMP Router Discovery Overview

Router discovery uses Internet Control Message Protocol (ICMP) router advertisements and router solicitation messages to allow a host to discover the addresses of operational routers on the subnet. Hosts must discover routers before they can send IP datagrams outside their subnet.

Router discovery allows a host to discover the addresses of operational routers on the subnet. The JUNOS Software implementation of router discovery supports server mode only.

Each router periodically multicasts a router advertisement from each of its multicast interfaces, announcing the IP address of that interface. Hosts listen for advertisements to discover the addresses of their neighboring routers. When a host starts, it can send a multicast router solicitation to ask for immediate advertisements.

The router discovery messages do not constitute a routing protocol. They enable hosts to discover the existence of neighboring routers, but do not determine which router is best to reach a particular destination.

This section discusses the following topics:

- Operation of a Router Discovery Server on page 659
- Router Advertisement Messages on page 660

Operation of a Router Discovery Server

The router discovery server distributes information about the addresses of all routers on directly connected networks and about their preferences for becoming the default router. (A host sends a packet to the default router if the host does not have a route to a destination in its routing table.) The server does this by periodically sending router advertisement packets out each interface on which router discovery is enabled. In addition to containing the router addresses, these packets also announce the existence of the server itself.

The server can either transmit broadcast or multicast router advertisement packets. Multicast packets are sent to 224.0.0.1, which is the all-hosts multicast address. When packets are sent to the all-hosts multicast address, or when an interface is configured for the limited-broadcast address 255.255.255.255, all IP addresses configured on the physical interface are included in the router advertisement. When the packets are being sent to a network or subnet broadcast address, only the address associated with that network or subnet is included in the router advertisement.

When the routing protocol process first starts on the server router, the server sends router advertisement packets every few seconds. Then, the server sends these packets less frequently, commonly every 10 minutes.

The server responds to route solicitation packets it receives from a client. The response is sent unicast unless a router advertisement packet is due to be sent out momentarily.

NOTE: The JUNOS Software does not support the ICMP router solicitation message with the source address as 0.0.0.0.

Router Advertisement Messages

Router advertisement messages include a preference level and a lifetime field for each advertised router address.

The preference level specifies the router's preference to become the default router. When a host chooses a default router address, it chooses the address with the highest preference. You can configure the preference level by including the priority statement as described in "Configuring the Addresses Included in ICMP Router Advertisements" on page 662.

The lifetime field indicates the maximum length of time that the advertised addresses are to be considered valid by hosts in the absence of further advertisements. You can configure the advertising rate by including the max-advertisement-interval and min-advertisement-interval statements, and you can configure the lifetime by including the lifetime statement. For configuration instructions, see "Configuring the Frequency of ICMP Router Advertisements" on page 663 and "Modifying the Lifetime in ICMP Router Advertisements" on page 663.

ICMP Router Discovery Standards

Router discovery is defined in RFC 1256, *ICMP Router Discovery Messages*.

To access Internet Requests for Comments (RFCs) and drafts, go to the Internet Engineering Task Force (IETF) Web site at http://www.ietf.org.

Chapter 27

ICMP Router Discovery Configuration Guidelines

This chapter describes the following tasks for configuring ICMP router discovery:

- Configuring ICMP Router Discovery on page 661
- Minimum ICMP Router Discovery Configuration on page 662
- Configuring the Addresses Included in ICMP Router Advertisements on page 662
- Configuring the Frequency of ICMP Router Advertisements on page 663
- Modifying the Lifetime in ICMP Router Advertisements on page 663
- Tracing ICMP Protocol Traffic on page 663

Configuring ICMP Router Discovery

To configure a router as a server for Internet Control Message Protocol (ICMP) router discovery, you can include the following statements in the configuration:

```
protocols {
    router-discovery {
    disable;
    traceoptions {
        file filename <files number> <size size> <world-readable | no-world-readable>;
        flag flag <detail> <disable>;
    }
    interface interface-name {
        min-advertisement-interval seconds;
        max-advertisement-interval seconds;
        lifetime seconds;
    }
    address address {
        (advertise | ignore);
        (broadcast | multicast);
        (priority number | ineligible);
    }
}
```

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for these statements.

By default, router discovery is disabled.

Minimum ICMP Router Discovery Configuration

To configure the router to be a router discovery server, you must include at least the following statement in the configuration. All other router discovery configuration statements are optional.

```
protocols {
    router-discovery;
}
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

NOTE: When you configure ICMP on an interface, you must also include the family inet statement at the [edit interfaces *interface-name* unit *logical-unit-number*] hierarchy level. For more information about the family inet statement, see the *JUNOS Network Interfaces Configuration Guide*.

Configuring the Addresses Included in ICMP Router Advertisements

To specify which addresses the router should include in its router advertisements, include the address statement:

```
address address {
    (advertise | ignore);
    (broadcast | multicast);
    (priority number | ineligible);
}
```

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for these statements.

Specify the IP address of the router, and optionally specify the following information about the router:

- Whether the server should include this address in its router advertisements—By default, the address is advertised. To disable this function, include the ignore statement.
- Whether the server should broadcast or multicast router advertisements—By default, advertisements are multicast if the router supports IP multicast; otherwise, they are broadcast. To modify the default functionality, include the broadcast or multicast statement.
- Preference of the address to become the default router—In the priority statement, a higher value for *number* indicates that the address has a greater preference for becoming the default router. The default value is 0, which means that the address has the least chance of becoming the default router. If the router at this address should never become the default router, include the ineligible statement. To modify the preference, include the preference statement. *number* can be a value in the range from 0 through 0x80000000. The default is 0.

Configuring the Frequency of ICMP Router Advertisements

The router discovery server sends router advertisement messages, which include route information and indicate to network hosts that the router is still operational. The server sends these messages periodically, with a time range defined by minimum and maximum values. By default, the server sends router advertisements every 400 to 600 seconds. To modify these times, include the min-advertisement-interval and max-advertisement-interval statements:

```
min-advertisement-interval seconds;
max-advertisement-interval seconds;
```

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for these statements.

Modifying the Lifetime in ICMP Router Advertisements

The lifetime field in router advertisement messages indicates how long a host should consider the advertised address to be valid. If this amount of time passes and the host has not received a router advertisement from the server, the route marks the advertised addresses as invalid. By default, addresses are considered to be valid for 1800 seconds (30 minutes).

To modify the router lifetime timer, include the lifetime statement:

```
lifetime seconds;
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

Tracing ICMP Protocol Traffic

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

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

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for these statements.

You can specify the following ICMP-specific options in the ICMP flag statement:

- error—Trace error packets.
- info—Trace information packets.

- router-discovery—Trace all ICMP packets.
- redirect—Trace redirect packets.

You can specify the following global flag options:

- all—Trace everything.
- general—Trace general events.
- normal—Trace normal events.
- policy—Trace policy processing.
- route—Trace routing information.
- state—Trace state transitions.
- task—Trace routing protocol task processing.
- timer—Trace routing protocol timer processing.

NOTE: Use the trace flags detail and all with caution. These flags may cause the CPU to become very busy.

For general information about tracing and global tracing options, see "Tracing Global Routing Protocol Operations" on page 129.

Example: Tracing ICMP Protocol Traffic

Trace only unusual or abnormal operations to a file called routing-log, and trace router discovery state transitions to a file called icmp-log:

```
[edit]
routing-options {
   traceoptions {
      file routing-log;
   }
}
protocols {
   router-discovery {
      traceoptions {
         file icmp-log;
         flag state;
      }
   }
}
```

Chapter 28

Summary of ICMP Router Discovery Configuration Statements

The following sections explain each of the Internet Control Message Protocol (ICMP) router discovery configuration statements. The statements are organized alphabetically.

address

Syntax

```
address address {
   (advertise | ignore);
   (broadcast | multicast);
   (priority number | ineligible);
}
```

Hierarchy Level [edit logical-systems *logical-system-name* protocols router-discovery], [edit protocols router-discovery]

Release Information Statement introduced before JUNOS Release 7.4.

Description IP addresses to include in router advertisement packets.

Options *address*—IP address. To specify more than one address, specify multiple addresses or include multiple **address** statements.

The remaining statements are explained separately.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics

- Configuring the Addresses Included in ICMP Router Advertisements on page 662

advertise

Syntax (advertise | ignore);

Hierarchy Level [edit logical-systems *logical-system-name* protocols router-discovery address *address*], [edit protocols router-discovery address *address*]

Release Information Statement introduced before JUNOS Release 7.4.

Description Specify whether the server should advertise the IP address in its router advertisement packets:

- advertise—Advertise the IP address in its router advertisement packets.
- ignore—Do not advertise the IP addresses in router advertisement packets.

Default advertise

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring the Addresses Included in ICMP Router Advertisements on page 662

broadcast

Syntax (broadcast | multicast);

Hierarchy Level [edit logical-systems *logical-system-name* protocols router-discovery address *address*], [edit protocols router-discovery address *address*]

Release Information Statement introduced before JUNOS Release 7.4.

Description Specify when the server should include the IP addresses in router advertisement packets. On the same physical interfaces, some addresses might be included only in multicast packets, while others might be included only in broadcast packets.

If you specify broadcast, the server includes the addresses in router advertisement packets only if the packets are broadcast.

Default multicast if the router supports IP multicast; broadcast if the router does not support IP multicast.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- multicast
- Configuring the Addresses Included in ICMP Router Advertisements on page 662

disable

Syntax disable;

Hierarchy Level [edit logical-systems *logical-system-name* protocols router-discovery],
[edit protocols router-discovery]

Release Information Statement introduced before JUNOS Release 7.4.

Description Disable router discovery.

Default The configured object is enabled (operational) unless explicitly disabled.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Minimum ICMP Router Discovery Configuration on page 662

ignore

See advertise

ineligible

See priority

interface

Syntax

```
interface interface-name {
    min-advertisement-interval seconds;
    max-advertisement-interval seconds;
    lifetime seconds;
}
```

Hierarchy Level [edit logical-systems *logical-system-name* protocols router-discovery], [edit protocols router-discovery]

Release Information Statement introduced before JUNOS Release 7.4.

Description Specify physical interfaces on which to configure timers for router advertisement messages.

Options *interface-name*—Name of an interface. Specify the full interface name, including the physical and logical address components. To configure all interfaces, specify **all**. For details about specifying interfaces, see the *JUNOS Network Interfaces Configuration Guide*.

The remaining statements are explained separately.

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

"Configuring the Frequency of ICMP Router Advertisements" on page 663

Related Topics

- Modifying the Lifetime in ICMP Router Advertisements on page 663

lifetime

Syntax lifetime *seconds*;

Hierarchy Level [edit logical-systems *logical-system-name* protocols router-discovery interface *interface-name*],
[edit protocols router-discovery interface *interface-name*]

Release Information Statement introduced before JUNOS Release 7.4.

Description How long the addresses sent by the server in its router advertisement packets are valid. This time must be long enough so that another router advertisement packet is sent before the lifetime has expired. The lifetime value is placed in the advertisement lifetime field of the router advertisement packet.

Options *seconds*—Lifetime value. A value of 0 indicates that one or more addresses are no longer valid.
Range: Three times the value set by the max-advertisement-interval statement through 2 hours, 30 minutes (9000 seconds)
Default: 1800 seconds (30 minutes, which is three times the default value for the max-advertisement-interval statement)

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- max-advertisement-interval
- Modifying the Lifetime in ICMP Router Advertisements on page 663

max-advertisement-interval

Syntax max-advertisement-interval *seconds*;

Hierarchy Level [edit logical-systems *logical-system-name* protocols router-discovery interface *interface-name*],
[edit protocols router-discovery interface *interface-name*]

Release Information Statement introduced before JUNOS Release 7.4.

Description Maximum time the router waits before sending periodic router advertisement packets out the interface. These packets are broadcast or multicast, depending on how the address corresponding to this physical interface is configured.

Options *seconds*—Maximum time between router advertisement packets.
Range: 4 through 1800 seconds
Default: 600 seconds (10 minutes)

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- broadcast
- lifetime
- min-advertisement-interval
- multicast
- Configuring the Frequency of ICMP Router Advertisements on page 663

min-advertisement-interval

Syntax min-advertisement-interval *seconds*;

Hierarchy Level [edit logical-systems *logical-system-name* protocols router-discovery interface *interface-name*],
[edit protocols router-discovery interface *interface-name*]

Release Information Statement introduced before JUNOS Release 7.4.

Description Minimum time the router waits before sending router advertisement packets out the interface in response to route solicitation packets it receives from a client. These packets are broadcast or multicast, depending on how the address corresponding to this physical interface is configured.

Options *seconds*—Minimum time between router advertisement packets.
Range: 3 seconds through 1800 seconds
Default: 400 seconds (0.75 times the default value for the max-advertisement-interval statement)

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- broadcast
- max-advertisement-interval
- multicast
- Configuring the Frequency of ICMP Router Advertisements on page 663

multicast

Syntax (multicast | broadcast);

Hierarchy Level [edit logical-systems *logical-system-name* protocols router-discovery address *address*], [edit protocols router-discovery address *address*]

Release Information Statement introduced before JUNOS Release 7.4.

Description Specify when the server should include the IP addresses in router advertisement packets. On the same physical interfaces, some addresses might be included only in multicast packets, while others might be included only in broadcast packets.

If you specify multicast, the server includes the addresses in router advertisement packets only if the packets are multicast. If the router supports IP multicast, and if the interface supports IP multicast, multicast is the default. Otherwise, the addresses are included in broadcast router advertisement packets. If the router does not support IP multicast, the addresses are not included.

Default multicast if the router supports IP multicast; broadcast if the router does not support IP multicast.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- broadcast
- Configuring the Addresses Included in ICMP Router Advertisements on page 662

priority

Syntax priority (*number* | ineligible);

Hierarchy Level [edit logical-systems *logical-system-name* protocols router-discovery address *address*], [edit protocols router-discovery address *address*]

Release Information Statement introduced before JUNOS Release 7.4.

Description Preference of the address to become a default router. This preference is set relative to the preferences of other router addresses on the same subnet.

Options ineligible—Address can never become the default router.

priority *number*—Preference of the addresses for becoming the default router. A higher value indicates that the address has a greater preference for becoming the default router.
Range: 0 through 0x80000000
Default: 0 (This address has the least chance of becoming the default router.)

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring the Addresses Included in ICMP Router Advertisements on page 662

router-discovery

Syntax router-discovery { ... }

Hierarchy Level [edit logical-systems *logical-system-name* protocols], [edit protocols]

Release Information Statement introduced before JUNOS Release 7.4.

Description Enable ICMP router discovery (server mode) on the router.

Default Router discovery is disabled on the router.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Minimum ICMP Router Discovery Configuration on page 662

Chapter 29

Introduction to Neighbor Discovery

This chapter discusses the following topics that provide background information about neighbor discovery:

- Neighbor Discovery Overview on page 675
- Neighbor Discovery Standards on page 676

Neighbor Discovery Overview

Neighbor discovery is a protocol that allows different nodes on the same link to advertise their existence to their neighbors, and to learn about the existence of their neighbors.

A router periodically multicasts a router advertisement from each of its multicast interfaces, announcing its availability. Hosts listen for these advertisements for address autoconfiguration and discovery of link-local addresses of the neighboring routers. When a host starts, it multicasts a router solicitation to ask for immediate advertisements.

The router discovery messages do not constitute a routing protocol. They enable hosts to discover the existence of neighboring routers, but are not used to determine which router is best to reach a particular destination.

Neighbor discovery uses the following Internet Control Message Protocol version 6 (ICMPv6) messages: router solicitation, router advertisement, neighbor solicitation, neighbor advertisement, and redirect.

Neighbor discovery for IPv6 replaces the following IPv4 protocols: router discovery (RDISC), Address Resolution Protocol (ARP), and ICMPv4 redirect.

JUNOS Release 9.3 and later supports Secure Neighbor Discovery (SEND). SEND enables you to secure Neighbor Discovery protocol (NDP) messages. It is applicable in environments where physical security on a link is not assured and attacks on NDP messages are a concern. The JUNOS Software secures NDP messages through cryptographically generated addresses (CGAs).

This section discusses the following topics:

- Router Discovery on page 676
- Address Resolution on page 676
- Redirect on page 676

Router Discovery

Router advertisements can contain a list of prefixes. These prefixes are used for address autoconfiguration, to maintain a database of onlink (on the same data link) prefixes, and for duplication address detection. If a node is onlink, the router forwards packets to that node. If the node is not onlink, the packets are sent to the next router for consideration. For IPv6, each prefix in the prefix list can contain a prefix length, a valid lifetime for the prefix, a preferred lifetime for the prefix, an onlink flag, and an autoconfiguration flag. This information enables address autoconfiguration and the setting of link parameters such as maximum transmission unit (MTU) size and hop limit.

Address Resolution

For IPv6, ICMPv6 neighbor discovery replaces Address Resolution Protocol (ARP) for resolving network addresses to link-level addresses. Neighbor discovery also handles changes in link-layer addresses, inbound load balancing, anycast addresses, and proxy advertisements.

Nodes requesting the link-layer address of a target node multicast a neighbor solicitation message with the target address. The target sends back a neighbor advertisement message containing its link-layer address.

Neighbor solicitation and advertisement messages are used for detecting duplicate unicast addresses on the same link. Autoconfiguration of an IP address depends on whether there is a duplicate address on that link. Duplicate address detection is a requirement for autoconfiguration.

Neighbor solicitation and advertisement messages are also used for neighbor unreachability detection. Neighbor unreachability detection involves detecting the presence of a target node on a given link.

Redirect

Redirect messages are sent to inform a host of a better next-hop router to a particular destination or an onlink neighbor. This is similar to ICMPv4 redirect.

Neighbor Discovery Standards

Neighbor discovery is defined in the following documents:

- RFC 4861, *Neighbor Discovery for IP Version 6*
- RFC 4862, *IPv6 Stateless Address Autoconfiguration*
- RFC 4884, *Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 Specification*

To access Internet RFCs and drafts, go to the Internet Engineering Task Force (IETF) website at http://www.ietf.org.

Chapter 30

Neighbor Discovery Configuration Guidelines

This chapter describes the following tasks for configuring and monitoring neighbor discovery router advertisement messages:

- Configuring Neighbor Discovery on page 677
- Minimum Neighbor Discovery Configuration on page 678
- Configuring an Interface to Send Neighbor Discovery Advertisements on page 678
- Configuring the Hop Count in Outgoing Neighbor Discovery Packets on page 679
- Configuring the Lifetime for the Default Neighbor Discovery Router on page 679
- Enabling Stateful Autoconfiguration with Neighbor Discovery on page 679
- Configuring the Frequency of Neighbor Discovery Advertisements on page 680
- Configuring the Delay Before Neighbor-Discovery Neighbors Mark the Router as Down on page 680
- Configuring the Frequency of Neighbor Solicitation Messages on page 681
- Configuring the Prefix Information Included in Neighbor Discovery Advertisements on page 681
- Tracing Neighbor Discovery Protocol Traffic on page 683

Configuring Neighbor Discovery

To configure neighbor discovery, include the following statements. You configure router advertisement on a per-interface basis.

```
protocols {
    router-advertisement {
    interface interface-name {
        current-hop-limit number;
        default-lifetime seconds;
        (managed-configuration |no-managed-configuration);
        max-advertisement-interval seconds;
        min-advertisement-interval seconds;
        (other-stateful-configuration | no-other-stateful-configuration);
        prefix prefix {
            (autonomous | no-autonomous);
            (on-link | no-on-link);
            preferred-lifetime seconds;
```

```
            valid-lifetime seconds;
        }
        reachable-time milliseconds;
        retransmit-timer milliseconds;
        traceoptions {
            file filename <files number> <size size> <world-readable | no-world-readable>;
            flag flag <detail> <disable>;
        }
    }
}
```

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for these statements.

Minimum Neighbor Discovery Configuration

To configure the router to send router advertisement messages, you must include at least the following statements in the configuration. All other router advertisement configuration statements are optional.

```
protocols {
    router-advertisement {
        interface interface-name {
            prefix prefix;
        }
    }
}
```

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for these statements.

NOTE: When you configure neighbor discovery router advertisement on an interface, you must also include the family inet6 statement at the [edit interfaces *interface-name* unit *logical-unit-number*] hierarchy level. For more information about the family inet6 statement, see the *JUNOS Network Interfaces Configuration Guide*.

Configuring an Interface to Send Neighbor Discovery Advertisements

To configure an interface to send router advertisement messages, include the interface statement:

```
interface interface-name;
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

Specify the interface name in the following format:

physical<:*channel*>.*logical*

For more information about interface names, see the *JUNOS Network Interfaces Configuration Guide*.

NOTE: The JUNOS Software enters the Neighbor Discovery Protocol (NDP) packets into the routing platform cache even if there is no known route to the source.

NOTE: If you are using Virtual Router Redundancy Protocol (VRRP) for IPv6, you must include the virtual-router-only statement on both the master and backup VRRP on the IPv6 router. For more information, see the *JUNOS High Availability Configuration Guide*.

Configuring the Hop Count in Outgoing Neighbor Discovery Packets

The current hop limit field in the router advertisement messages indicates the default value placed in the hop count field of the IP header for outgoing packets. To configure the hop limit, include the current-hop-limit statement:

```
current-hop-limit number;
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

The default hop limit is 64.

Configuring the Lifetime for the Default Neighbor Discovery Router

The default lifetime in router advertisement messages indicates the lifetime associated with the default router. To modify the default lifetime timer, include the default-lifetime statement:

```
default-lifetime seconds;
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

By default, the default router lifetime is three times the maximum advertisement interval. For more information about the maximum advertisement interval, see “Configuring the Frequency of Neighbor Discovery Advertisements” on page 680.

Enabling Stateful Autoconfiguration with Neighbor Discovery

You can set two fields in the router advertisement message to enable stateful autoconfiguration on a host: the managed configuration field and the other stateful configuration field. Setting the managed configuration field enables the host to use a stateful autoconfiguration protocol for address autoconfiguration, along with any stateless autoconfiguration already configured. Setting the other stateful configuration field enables autoconfiguration of other nonaddress-related information.

By default, stateful autoconfiguration is disabled.

To set the managed configuration field and enable address autoconfiguration, include the managed-configuration statement:

```
managed-configuration;
```

To disable managed configuration field, include the no-managed-configuration statement:

```
nomanaged-configuration;
```

To set the other stateful configuration field and enable autoconfiguration of other types of information, include the other-stateful-configuration statement:

```
other-stateful-configuration;
```

To disable other stateful configuration, include the no-other-stateful-configuration statement:

```
no-other-stateful-configuration;
```

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for these statements.

Configuring the Frequency of Neighbor Discovery Advertisements

The router sends router advertisements on each interface configured to transmit messages. The advertisements include route information and indicate to network hosts that the router is operational. The router sends these messages periodically, with a time range defined by minimum and maximum values.

To modify the router advertisement interval, include the min-advertisement-interval and max-advertisement-interval statements:

```
min-advertisement-interval seconds;
max-advertisement-interval seconds;
```

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for these statements.

By default, the maximum advertisement interval is 600 seconds and the minimum advertisement interval is one-third the maximum interval, or 200 seconds.

Configuring the Delay Before Neighbor-Discovery Neighbors Mark the Router as Down

After receiving a reachability confirmation from a neighbor, a node considers that neighbor reachable for a certain amount of time without receiving another confirmation. This mechanism is used for neighbor unreachability detection, a mechanism for finding link failures to a target node.

To modify the reachable time limit, include the reachable-time statement:

reachable-time *milliseconds*;

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

By default, the reachable time period is 0 milliseconds.

Configuring the Frequency of Neighbor Solicitation Messages

The retransmit timer determines the retransmission frequency of neighbor solicitation messages. This timer is used to detect when a neighbor has become unreachable and to resolve addresses. To modify the retransmit timer, include the retransmit-timer statement:

retransmit-timer *milliseconds*;

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

By default, the retransmit timer is 0 milliseconds.

Configuring the Prefix Information Included in Neighbor Discovery Advertisements

Router advertisement messages carry prefixes and information about them. A prefix is onlink when it is assigned to an interface on a specified link. The prefixes specify whether they are onlink or not onlink. A node considers a prefix to be onlink if it is represented by one of the link's prefixes, a neighboring router specifies the address as the target of a redirect message, a neighbor advertisement message is received for the (target) address, or any neighbor discovery message is received from the address. These prefixes are also used for address autoconfiguration. The information about the prefixes specifies the lifetime of the prefixes, whether the prefix is autonomous, and whether the prefix is onlink.

You can perform the following tasks when configuring the prefix information:

- Setting the Prefix for Onlink Determination on page 681
- Setting the Prefix for Stateless Address Autoconfiguration on page 682
- Configuring the Preferred Lifetime on page 682
- Configuring the Valid Lifetime on page 682

Setting the Prefix for Onlink Determination

You can specify prefixes in the router advertisement messages as onlink. When set as onlink, the prefixes are used for onlink determination. By default, prefixes are onlink.

To explicitly set prefixes as onlink, include the on-link statement:

on-link;

To set prefixes as not onlink, include the no-on-link statement:

```
no-on-link;
```

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for these statements.

Setting the Prefix for Stateless Address Autoconfiguration

You can specify prefixes in the router advertisement messages as autonomous. When set as autonomous, the prefixes are used for stateless address autoconfiguration. By default, prefixes are autonomous.

To explicitly specify prefixes as autonomous, include the autonomous statement:

```
autonomous;
```

To specify prefixes as not autonomous, include the no-autonomous statement:

```
no-autonomous;
```

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for these statements.

Configuring the Preferred Lifetime

The preferred lifetime for the prefixes in the router advertisement messages specifies how long the prefix generated by stateless autoconfiguration remains preferred. By default, the preferred lifetime is set to 604,800 seconds.

To configure the preferred lifetime, include the preferred-lifetime statement:

```
preferred-lifetime seconds;
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

If you set the preferred lifetime to 0xffffffff, the lifetime is infinite.

The preferred lifetime value must never exceed the valid lifetime value.

Configuring the Valid Lifetime

The valid lifetime for the prefixes in the router advertisement messages specifies how long the prefix remains valid for onlink determination. By default, the valid lifetime is set to 2,592,000 seconds.

To configure the valid lifetime, include the valid-lifetime statement:

```
valid-lifetime seconds;
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

If you set the valid lifetime to 0xffffffff, the lifetime is infinite.

The valid lifetime value must never be smaller than the preferred lifetime value.

Tracing Neighbor Discovery Protocol Traffic

You can trace various Neighbor Discovery protocol traffic to help debug Neighbor Discovery protocol issues. To trace Neighbor Discovery protocol traffic include the traceoptions statement at the [edit protocols router-advertisement] hierarchy level:

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

Global tracing options are inherited from the configuration set by the traceoptions statement at the [edit routing-options] hierarchy level. You can override the following global trace options for the Neighbor Discovery protocol using the traceoptions flag statement included at the [edit protocols router-advertisement] hierarchy level:

- all—All tracing operations
- general—All normal operations and routing table changes (a combination of the normal and route trace operations)
- normal—Normal events
- policy—Policy processing
- route—Routing information
- state—State transitions
- task—Routing protocol task processing
- timer—Routing protocol timer processing

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

Related Topics

- traceoptions statement
- For more information about tracing and global tracing options, see Tracing Global Routing Protocol Operations on page 129.

Chapter 31

Summary of Neighbor Discovery Router Advertisement Configuration Statements

The following sections explain each of the neighbor discovery router advertisement configuration statements. The statements are organized alphabetically.

autonomous

Syntax (autonomous | no-autonomous);

Hierarchy Level [edit logical-systems *logical-system-name* protocols router-advertisement interface *interface-name* prefix *prefix*],
[edit protocols router-advertisement interface *interface-name* prefix *prefix*]

Release Information Statement introduced before JUNOS Release 7.4.

Description Specify whether prefixes in the router advertisement messages are used for stateless address autoconfiguration:

- autonomous—Use prefixes for address autoconfiguration.
- no-autonomous—Do not use prefixes for address autoconfiguration.

Default autonomous

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Setting the Prefix for Stateless Address Autoconfiguration on page 682

current-hop-limit

Syntax current-hop-limit *number*;

Hierarchy Level [edit logical-systems *logical-system-name* protocols router-advertisement interface *interface-name*],
[edit protocols router-advertisement interface *interface-name*]

Release Information Statement introduced before JUNOS Release 7.4.

Description Default value placed in the hop count field of the IP header for outgoing packets.

Options *number*—Hop limit. A value of 0 means the limit is unspecified by this router.
Range: 0 through 255
Default: 64

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring the Hop Count in Outgoing Neighbor Discovery Packets on page 679

default-lifetime

Syntax default-lifetime *seconds*;

Hierarchy Level [edit logical-systems *logical-system-name* protocols router-advertisement interface *interface-name*],
[edit protocols router-advertisement interface *interface-name*]

Release Information Statement introduced before JUNOS Release 7.4.

Description Lifetime associated with a default router.

Options *seconds*—Default lifetime. A value of 0 means this router is not the default router.
Range: Maximum advertisement interval value through 9000 seconds
Default: Three times the maximum advertisement interval value

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- max-advertisement-interval
- Configuring the Lifetime for the Default Neighbor Discovery Router on page 679

interface

Syntax

```
interface interface-name {
    current-hop-limit number;
    default-lifetime seconds;
    (managed-configuration | no-managed-configuration);
    max-advertisement-interval seconds;
    min-advertisement-interval seconds;
    (other-stateful-configuration | no-other-stateful-configuration);
    prefix prefix {
        (autonomous | no-autonomous);
        (on-link | no-on-link);
        preferred-lifetime seconds;
        valid-lifetime seconds;
    }
    reachable-time milliseconds;
    retransmit-timer milliseconds;
}
```

Hierarchy Level [edit logical-systems *logical-system-name* protocols router-advertisement],
[edit protocols router-advertisement]

Release Information Statement introduced before JUNOS Release 7.4.

Description Configure router advertisement properties on an interface. To configure more than one interface, include the interface statement multiple times.

Options *interface-name*—Name of an interface. Specify the full interface name, including the physical and logical address components.

The remaining statements are explained separately.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics

- Configuring an Interface to Send Neighbor Discovery Advertisements on page 678

managed-configuration

Syntax (managed-configuration | no-managed-configuration);

Hierarchy Level [edit logical-systems *logical-system-name* protocols router-advertisement interface *interface-name*],
[edit protocols router-advertisement interface *interface-name*]

Release Information Statement introduced before JUNOS Release 7.4.

Description Specify whether to enable the host to use a stateful autoconfiguration protocol for address autoconfiguration, along with any stateless autoconfiguration already configured:

- managed-configuration—Enable host to use stateful autoconfiguration.
- no-managed-configuration—Disable host from using stateful autoconfiguration.

Default The configured object is disabled unless explicitly enabled.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Enabling Stateful Autoconfiguration with Neighbor Discovery on page 679

max-advertisement-interval

Syntax max-advertisement-interval *seconds*;

Hierarchy Level [edit logical-systems *logical-system-name* protocols router-advertisement interface *interface-name*],
[edit protocols router-advertisement interface *interface-name*]

Release Information Statement introduced before JUNOS Release 7.4.

Description Maximum interval between each router advertisement message.

Options *seconds*—Maximum interval.
Range: 4 through 1800 seconds
Default: 600 seconds

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- min-advertisement-interval
- Configuring the Frequency of Neighbor Discovery Advertisements on page 680

min-advertisement-interval

Syntax min-advertisement-interval *seconds*;

Hierarchy Level [edit logical-systems *logical-system-name* protocols router-advertisement interface *interface-name*],
[edit protocols router-advertisement interface *interface-name*]

Release Information Statement introduced before JUNOS Release 7.4.

Description Minimum interval between each router advertisement message.

Options *seconds*—Minimum interval.
Range: 3 seconds through three-quarter times the maximum advertisement interval value
Default: One-third the maximum advertisement interval value

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- max-advertisement-interval
- Configuring the Frequency of Neighbor Discovery Advertisements on page 680

no-autonomous

See autonomous

no-managed-configuration

See managed-configuration

no-on-link

See on-link

no-other-stateful-configuration

See other-stateful-configuration

on-link

Syntax (on-link | no-on-link);

Hierarchy Level [edit logical-systems *logical-system-name* protocols router-advertisement interface *interface-name* prefix *prefix*],
[edit protocols router-advertisement interface *interface-name* prefix *prefix*]

Release Information Statement introduced before JUNOS Release 7.4.

Description Specify whether to enable prefixes to be used for onlink determination:

- no-on-link—Disable prefixes from being used for onlink determination.
- on-link—Enable prefixes to be used for onlink determination.

Default The configured object is enabled unless explicitly disabled.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring the Prefix Information Included in Neighbor Discovery Advertisements on page 681

other-stateful-configuration

Syntax (other-stateful-configuration | no-other-stateful-configuration);

Hierarchy Level [edit logical-systems *logical-system-name* protocols router-advertisement interface *interface-name*],
[edit protocols router-advertisement interface *interface-name*]

Release Information Statement introduced before JUNOS Release 7.4.

Description Specify whether to enable autoconfiguration of other nonaddress-related information:

- no-other-stateful-configuration—Disable autoconfiguration of other nonaddress-related information.
- other-stateful-configuration—Enable autoconfiguration of other nonaddress-related information.

Default The configured object is disabled unless explicitly enabled.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Enabling Stateful Autoconfiguration with Neighbor Discovery on page 679

preferred-lifetime

Syntax preferred-lifetime *seconds*;

Hierarchy Level [edit logical-systems *logical-system-name* protocols router-advertisement interface *interface-name* prefix *prefix*],
[edit protocols router-advertisement interface *interface-name* prefix *prefix*]

Release Information Statement introduced before JUNOS Release 7.4.

Description Specify how long the prefix generated by stateless autoconfiguration remains preferred.

Options *seconds*—Preferred lifetime, in seconds. If you set the preferred lifetime to 0xffffffff, the lifetime is infinite. The preferred lifetime is never greater than the valid lifetime.
Default: 604,800 seconds

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- valid-lifetime
- Configuring the Preferred Lifetime on page 682

prefix

Syntax
```
prefix prefix {
    (autonomous | no-autonomous);
    (on-link | no-on-link);
    preferred-lifetime seconds;
    valid-lifetime seconds;
}
```

Hierarchy Level [edit logical-systems *logical-system-name* protocols router-advertisement interface *interface-name*],
[edit protocols router-advertisement interface *interface-name*]

Release Information Statement introduced before JUNOS Release 7.4.

Description Configure prefix properties in router advertisement messages.

Options *prefix*—Prefix name.

The remaining statements are explained separately.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring the Prefix Information Included in Neighbor Discovery Advertisements on page 681

reachable-time

Syntax
```
reachable-time milliseconds;
```

Hierarchy Level [edit logical-systems *logical-system-name* protocols router-advertisement interface *interface-name*],
[edit protocols router-advertisement interface *interface-name*]

Release Information Statement introduced before JUNOS Release 7.4.

Description Set the length of time that a node considers a neighbor reachable until another reachability confirmation is received from that neighbor.

Options *milliseconds*—Reachability time limit.
Range: 0 through 3,600,000 milliseconds
Default: 0 milliseconds

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring the Delay Before Neighbor-Discovery Neighbors Mark the Router as Down on page 680

retransmit-timer

Syntax retransmit-timer *milliseconds*;

Hierarchy Level [edit logical-systems *logical-system-name* protocols router-advertisement interface *interface-name*],
[edit protocols router-advertisement interface *interface-name*]

Release Information Statement introduced before JUNOS Release 7.4.

Description Set the retransmission frequency of neighbor solicitation messages.

Options *milliseconds*—Retransmission frequency.
Default: 0 milliseconds

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring the Frequency of Neighbor Solicitation Messages on page 681

router-advertisement

Syntax router-advertisement {...}

Hierarchy Level [edit logical-systems *logical-system-name* protocols],
[edit protocols]

Release Information Statement introduced before JUNOS Release 7.4.

Description Enable router advertisement.

The remaining statements are explained separately.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring an Interface to Send Neighbor Discovery Advertisements on page 678

traceoptions

Syntax

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

Hierarchy Level [edit logical-systems *logical-system-name* protocols router-advertisement], [edit protocols router-advertisement]

Release Information Statement introduced before JUNOS Release 7.4.

Description Specify router advertisement protocol-level tracing options.

Default The default trace options are inherited from the global traceoptions statement.

Options

disable—(Optional) Disable the tracing operation. One use of this option is to disable a single operation when you have defined a broad group of tracing operations, such as all.

file *filename*—Name of the file to receive the output of the tracing operation. Enclose the name in quotation marks. We recommend that you place router advertisement tracing output in the file /var/log/router-advertisement-log.

files *number*—(Optional) Maximum number of trace files. When a trace file named *trace-file* reaches its maximum size, it is renamed *trace-file*.0, then *trace-file*.1, and so on, until the maximum number of trace files is reached. Then, the oldest trace file is overwritten. If you specify a maximum number of files, you must also specify a maximum file size with the size option.
Range: 2 through 1000 files
Default: 10 files

flag *flag*—Tracing operation to perform. To specify more than one tracing operation, include multiple flag statements.

- all—All tracing operations
- general—A combination of the normal and route trace operations
- normal—All normal operations.

  Default: If you do not specify this option, only unusual or abnormal operations are traced.
- policy—Policy operations and actions
- route—Routing table changes
- state—State transitions
- task—Interface transactions and processing
- timer—Timer usage

no-world-readable—(Optional) Prevent any user from reading the log file.

size *size*—(Optional) Maximum size of each trace file, in kilobytes (KB) or megabytes (MB). When a trace file named *trace-file* reaches this size, it is renamed *trace-file*.0. When the *trace-file* again reaches its maximum size, *trace-file*.0 is renamed *trace-file*.1 and *trace-file* is renamed *trace-file*.0. This renaming scheme continues until the maximum number of trace files is reached. Then, the oldest trace file is overwritten. If you specify a maximum file size, you must also specify a maximum number of trace files with the files option.
Syntax: *x*k to specify KB, *x*m to specify MB, or *x*g to specify GB
Range: 10 KB through the maximum file size supported on your system
Default: 128 KB

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

Required Privilege Level

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

Related Topics ■ Tracing Neighbor Discovery Protocol Traffic on page 683

traceoptions

Syntax

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

Hierarchy Level [edit logical-systems *logical-system-name* protocols router-advertisement],
[edit protocols router-advertisement]

Release Information Statement introduced before JUNOS Release 7.4.

Description Specify router advertisement protocol-level tracing options.

Default The default trace options are inherited from the global traceoptions statement.

Options disable—(Optional) Disable the tracing operation. One use of this option is to disable a single operation when you have defined a broad group of tracing operations, such as all.

file *filename*—Name of the file to receive the output of the tracing operation. Enclose the name in quotation marks. We recommend that you place router advertisement tracing output in the file /var/log/router-advertisement-log.

files *number*—(Optional) Maximum number of trace files. When a trace file named *trace-file* reaches its maximum size, it is renamed *trace-file*.0, then *trace-file*.1, and so on, until the maximum number of trace files is reached. Then, the oldest trace file is overwritten. If you specify a maximum number of files, you must also specify a maximum file size with the size option.
Range: 2 through 1000 files
Default: 10 files

flag *flag*—Tracing operation to perform. To specify more than one tracing operation, include multiple flag statements.

- all—All tracing operations
- general—A combination of the normal and route trace operations
- normal—All normal operations.

  Default: If you do not specify this option, only unusual or abnormal operations are traced.
- policy—Policy operations and actions
- route—Routing table changes
- state—State transitions
- task—Interface transactions and processing
- timer—Timer usage

no-world-readable—(Optional) Prevent any user from reading the log file.

size *size*—(Optional) Maximum size of each trace file, in kilobytes (KB) or megabytes (MB). When a trace file named *trace-file* reaches this size, it is renamed *trace-file*.0. When the *trace-file* again reaches its maximum size, *trace-file*.0 is renamed *trace-file*.1 and *trace-file* is renamed *trace-file*.0. This renaming scheme continues until the maximum number of trace files is reached. Then, the oldest trace file is overwritten. If you specify a maximum file size, you must also specify a maximum number of trace files with the files option.
Syntax: *x*k to specify KB, *x*m to specify MB, or *x*g to specify GB
Range: 10 KB through the maximum file size supported on your system
Default: 128 KB

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

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Tracing Neighbor Discovery Protocol Traffic on page 683

valid-lifetime

Syntax valid-lifetime *seconds*;

Hierarchy Level [edit logical-systems *logical-system-name* protocols router-advertisement interface *interface-name* prefix *prefix*],
[edit protocols router-advertisement interface *interface-name* prefix *prefix*]

Release Information Statement introduced before JUNOS Release 7.4.

Description Specify how long the prefix remains valid for onlink determination.

Options *seconds*—Valid lifetime, in seconds. If you set the valid lifetime to 0xffffffff, the lifetime is infinite.
Default: 2,592,000 seconds

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- preferred-lifetime
- Configuring the Valid Lifetime on page 682

Chapter 32

Secure Neighbor Discovery Configuration Guidelines

This chapter discusses the following topics that describe how to configure Secure Neighbor Discovery:

- Secure Neighbor Discovery Configuration Overview on page 699
- Configuring Secure Neighbor Discovery on page 699
- Enabling Secure Neighbor Discovery on page 700
- Configuring Cryptographically Generated Addresses for Secure Neighbor Discovery on page 700
- Configuring Timestamps for Secure Neighbor Discovery on page 701
- Tracing Secure Neighbor Discovery Protocol Traffic on page 702

Secure Neighbor Discovery Configuration Overview

The Secure Neighbor Discovery (SEND) Protocol provides support for protecting Neighbor Discovery Protocol (NDP) messages. SEND is applicable in environments where physical security on a link is not ensured and attacks on NDP messages are a concern. The JUNOS Software implementation secures NDP messages through cryptographically generated addresses (CGAs).

You must also enable IPv6 on at least one interface. Because SEND relies on dynamically generated CGAs, it does not support static IPv6 addresses.

Configuring Secure Neighbor Discovery

To configure Secure Neighbor Discovery (SEND), include the following statements:

```
protocols {
    neighbor-discovery {
        secure {
            security-level {
                (default | secure-messages-only);
            }
            cryptographic-address {
                key-length number;
                key-pair pathname;
            }
```

```
        timestamp {
            clock-drift number;
            known-peer-window seconds;
            new-peer-window seconds;
        }
        traceoptions {
            file <filename> <files number> <match regular-expression> <size size>
                <world-readable | no-world-readable>;
            flag flag;
            no-remote-trace;
        }
    }
  }
}
```

Enabling Secure Neighbor Discovery

To enable Secure Neighbor Discovery (SEND), include the following statements:

```
protocols {
    neighbor-discovery {
        secure {
            security-level {
                (default | secure-messages-only);
            }
        }
    }
}
```

Specify default to send and receive both secure and unsecured Neighbor Discovery Protocol (NDP) packets. To configure SEND to accept secured NDP messages only and to drop unsecured ones. specify secure-messages-only.

Configuring Cryptographically Generated Addresses for Secure Neighbor Discovery

The Secure Neighbor Discovery (SEND) Protocol uses cryptographically generated addresses (CGAs), as defined in RFC 3972, *Cryptographically Generated Addresses*, to ensure that the sender of a Neighbor Discovery Protocol (NDP) message is the "owner" of the claimed address. Each node must generate a public-private key pair before it can claim an address. The CGA is included in all outgoing neighbor solicitation and neighbor advertisement messages.

To configure parameters for CGAs, include the following statements:

```
protocols {
    neighbor-discovery {
        secure {
            cryptographic-address {
                key-length number;
                key-pair pathname;
            }
        }
    }
```

```
}
```

For information about how to configure parameters for cryptographic addresses, see the following sections:

- Specifying the Pathname for the Key File on page 701
- Specifying the RSA Key Length on page 701

Specifying the Pathname for the Key File

A cryptographic address is dynamically generated based on a public key and a subnet prefix. The private-public key file that is generated is placed by default in the /var/etc/rsa_key directory. You can a specify a pathname for that file. Include the key-pair *pathname* statement:

```
key-pair pathname;
```

For a complete list of hierarchy levels at which you can configure this statement, see the statement summary section for this statement.

Specifying the RSA Key Length

You can specify the length of the RSA key used to generate the CGA public-private pair. The default is 1024 bits, and you can specify a value from 1024 through 2048. Include the key-length *number* statement:

```
key-length number;
```

For a complete list of hierarchy levels at which you can configure this statement, see the statement summary section for this statement.

Configuring Timestamps for Secure Neighbor Discovery

The Secure Neighbor Discovery (SEND) Protocol supports several timestamp options, which are used to ensure that unsolicited solicitation and redirect messages are not being replayed. To configure timestamp parameters, include the following statements:

```
protocols {
    neighbor-discovery {
        secure {
            timestamp {
                new-peer-window seconds;
                known-peer-window seconds;
                clock-drift value;
            }
        }
    }
}
```

Use the new-peer-window *seconds* statement to specify the maximum allowable difference in the amount of time between the timestamp of a SEND message from a new peer and when it can be accepted. The default is 300 seconds.

Use the known-peer-window *seconds* statement to specify the expected interval between subsequent incoming SEND messages. The default is 1 second. A message from a known peer that arrives after the specified interval is discarded.

Use the clock drift *value* statement to specify a fractional value of 100 for the allowable drift in time between the synchronization of peers. The default is 0.01, or 1 percent.

Tracing Secure Neighbor Discovery Protocol Traffic

You can trace Secure Neighbor Discovery protocol traffic to help debug Secure Neighbor Discovery protocols issues. To trace Secure Neighbor Discovery protocol traffic include the traceoptions statement at the [edit protocols neighbor-discovery secure] hierarchy level:

```
traceoptions {
    file <filename> <files number> <match regular-expression> <size size>
        <world-readable | no-world-readable>;
    flag flag;
    no-remote-trace;
}
```

You can specify the following Secure Neighbor Discovery protocol-specific trace options using the flag statement:

- configuration—All configuration events
- cryptographic-address—Cryptographically generated address events
- protocol—Protocol processing events
- rsa—RSA events

Global tracing options are inherited from the configuration set by the traceoptions statement at the [edit routing-options] hierarchy level. You can override the following global trace options for the IS-IS protocol using the traceoptions flag statement included at the [edit protocols neighbor-discovery secure] hierarchy level:

- all—All tracing operations

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

Related Topics

- traceoptions
- Tracing Global Routing Protocol Operations on page 129.

Chapter 33

Summary of Secure Neighbor Discovery Configuration Statements

The following sections explain each of the Secure Neighbor Discovery configuration statements. The statements are organized alphabetically.

cryptographic-address

Syntax

```
cryptographic-address {
    key-length number;
    key-pair pathname;
}
```

Hierarchy Level [edit protocols neighbor-discovery secure]

Release Information Statement introduced in JUNOS Release 9.3.

Description Configure parameters for cryptographically generated addresses for Secure Neighbor Discovery.

The remaining statements are explained separately.

Required Privilege Level routing level—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics

- Configuring Cryptographically Generated Addresses for Secure Neighbor Discovery on page 700

key-length

Syntax key-length *number* {

Hierarchy Level [edit protocols neighbor-discovery secure cryptographic-address]

Release Information Statement introduced in JUNOS Release 9.3.

Description Specify the length of the RSA key used to generate the public-private key pair for the cryptographic address.

Default 1024

Options *number*—RSA key length.
Range: 1024 through 2048

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Specifying the RSA Key Length on page 701

key-pair

Syntax key-pair *pathname*;

Hierarchy Level [edit protocols neighbor-discovery secure cryptographic-address]

Release Information Statement introduced in JUNOS Release 9.3.

Description Specify the directory path of the public-private key file generated for the cryptographic address.

Options *pathname*—Directory path of the public-private key file. The default location of the file is /var/etc/rsa_key directory.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Specifying the Pathname for the Key File on page 701

neighbor-discovery

Syntax

```
neighbor-discovery {
    secure {
        security-level {
            (default | secure-messages-only);
        }
        cryptographic-address {
            key-length number;
            key-pair pathname;
        }
        timestamp {
            clock-drift number;
            known-peer-window number;
            new-peer-window number;
        }
        traceoptions {
            file <filename> <files number> <match regular-expression> <size size>
                <world-readable | no-world-readable>;
            flag flag;
            no-remote-trace;
        }
    }
}
```

Hierarchy Level [edit protocols]

Release Information Statement introduced in JUNOS Release 9.3.

Description Enable Secure Neighbor Discovery.

The remaining statements are explained separately.

Default Disabled

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics

- Enabling Secure Neighbor Discovery on page 700

secure

Syntax

```
secure {
   security-level {
      (default | secure-messages-only);
   }
   cryptographic-address {
      key-length number;
      key-pair pathname;
   }
   timestamp {
      clock-drift number;
      known-peer-window seconds;
      new-peer-window seconds;
   }
   traceoptions {
      file <filename> <files number> <match regular-expression> <size size>
         <world-readable | no-world-readable>;
      flag flag;
      no-remote-trace;
   }
}
```

Hierarchy Level [edit protocols neighbor-discovery]

Release Information Statement introduced in JUNOS Release 9.3.

Description Configure parameters for Secure Neighbor Discovery.

The remaining statements are explained separately.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics

- Configuring Cryptographically Generated Addresses for Secure Neighbor Discovery on page 700
- Configuring Timestamps for Secure Neighbor Discovery on page 701
- Tracing Secure Neighbor Discovery Protocol Traffic on page 702

security-level

Syntax

```
security-level {
   (default | secure-messages-only);
}
```

Hierarchy Level [edit protocols neighbor-discovery secure]

Release Information Statement introduced in JUNOS Release 9.3.

Description Configure the type of security mode for Secure Neighbor Discovery.

Options default—Accept and transmit both secure and unsecured messages.

secure-messages-only—Accept secure messages only. Discard unsecured messages.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics

- Enabling Secure Neighbor Discovery on page 700

timestamp

Syntax

```
timestamp {
    clock-drift value;
    known-peer-window seconds;
    new-peer-window seconds;
}
```

Hierarchy Level [edit protocols neighbor-discovery secure]

Release Information Statement introduced in JUNOS Release 9.3.

Description Configure timestamp options, which are used to ensure that solicitation and redirect messages are not being replayed.

Options

clock-drift *value*—Specify the allowable drift in time between the synchronization of peers. For *value*, specify a fractional value of 100.
Default: 0.01

known-peer-window *seconds*—Specify the expected interval in seconds between Secure Neighbor Discovery messages from an established peer.
Default: 1 second

new-peer-window *seconds*—Specify the maximum allowable time in seconds between the timestamp of a Secure Neighbor Discovery message from a new peer and when it can be accepted.
Default: 300 seconds

Required Privilege Level

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

Related Topics

- Configuring Timestamps for Secure Neighbor Discovery on page 701

traceoptions

Syntax

```
traceoptions {
    file <filename> <files number> <match regular-expression> <size size> <world-readable |
        no-world-readable>;
    flag flag;
    no-remote-trace;
}
```

Hierarchy Level [edit protocols neighbor-discovery secure]

Release Information Statement introduced in JUNOS Release 9.3.

Description Configure tracing operations for Secure Neighbor Discovery events. To specify more than one tracing operation, include multiple flag statements.

Options

file *filename*—Name of the file to receive the tracing operation. Enclose the name within quotation marks. All files are placed in the directory /var/log.

files *number*—(Optional) Maximum number of trace files. When a trace file named *trace-file* reaches its maximum size, it is renamed *trace-file*.0, then *trace-file*.1 and so on, until the maximum number of trace files is reached. Then the oldest trace file is overwritten.
If you specify a maximum number of files, you must also specify a maximum file size with the size option.
Range: 2 through 1000 files
Default: 10 files

flag—Tracing operation to perform. To specify more than one tracing operation, include multiple flag statements.

Secure Neighbor Discovery Tracing Options

- configuration—All configuration events.
- cryptographic-address—Cryptographically generated address events.
- protocol—All protocol processing events.
- rsa—RSA events.

Global Tracing Options

- all—All tracing operations.

You can specify one or more of following flag modifiers:

- detail—Provide detailed trace information
- receive—Packets being received
- send—Packets being transmitted

match—(Optional) Specify a regular expression to match the output of the trace file you want to log.

no-remote-trace—Disable remote tracing globally or for a specific tracing operation.

no-world-readable—(Optional) Prevent any user from reading this log file.

size *size*—(Optional) Maximum size of each trace file, in kilobytes (KB) or megabytes (MB). When a trace file named *trace-file* reaches this size, it is renamed *trace-file*.0. When the *trace-file* again reaches its maximum size, *trace-file*.0 is renamed *trace-file*.1, and *trace-file* is renamed *trace-file*.0. This renaming scheme continues until the maximum number of trace files is reached. Then the oldest trace file is overwritten.
Syntax: *x*k to specify KB, *x*m to specify MB, or *x*g to specify GB
Range: 10 KB through the maximum file size supported on your system
Default: 128 KB

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

Required Privilege Level routing and trace—To view this statement in the configuration.
routing-control and trace-control—To add this statement to the configuration.

Related Topics
- Tracing Secure Neighbor Discovery Protocol Traffic on page 702

Part 6
BGP

- Introduction to BGP on page 713
- BGP Configuration Guidelines on page 721
- Summary of BGP Configuration Statements on page 803

Chapter 34

Introduction to BGP

This chapter discusses the following topics that provide background information about BGP:

- BGP Overview on page 714
- BGP Routes Overview on page 715
- Overview of BGP Messages on page 716
- BGP Standards on page 718

BGP Overview

BGP is an exterior gateway protocol (EGP) that is used to exchange routing information among routers in different autonomous systems (ASs). BGP routing information includes the complete route to each destination. BGP uses the routing information to maintain a database of network reachability information, which it exchanges with other BGP systems. BGP uses the network reachability information to construct a graph of AS connectivity, thus allowing BGP to remove routing loops and enforce policy decisions at the AS level.

Multiprotocol BGP (MBGP) extensions enable BGP to support IP version 6 (IPv6). MBGP defines the attributes MP_REACH_NLRI and MP_UNREACH_NLRI, which are used to carry IPv6 reachability information. Network layer reachability information (NLRI) update messages carry IPv6 address prefixes of feasible routes.

BGP allows for policy-based routing. You can use routing policies to choose among multiple paths to a destination and to control the redistribution of routing information.

BGP uses TCP as its transport protocol, using port 179 for establishing connections. Running over a reliable transport protocol eliminates the need for BGP to implement update fragmentation, retransmission, acknowledgment, and sequencing.

The JUNOS routing protocol software supports BGP version 4. This version of BGP adds support for Classless Interdomain Routing (CIDR), which eliminates the concept of network classes. Instead of assuming which bits of an address represent the network by looking at the first octet, CIDR allows you to explicitly specify the number of bits in the network address, thus providing a means to decrease the size of the routing tables. BGP version 4 also supports aggregation of routes, including the aggregation of AS paths.

This section discusses the following topics:

- Autonomous Systems on page 714
- AS Paths and Attributes on page 714
- External and Internal BGP on page 715

Autonomous Systems

An *autonomous system* (AS) is a set of routers that are under a single technical administration and normally use a single interior gateway protocol and a common set of metrics to propagate routing information within the set of routers. To other ASs, an AS appears to have a single, coherent interior routing plan and presents a consistent picture of what destinations are reachable through it.

AS Paths and Attributes

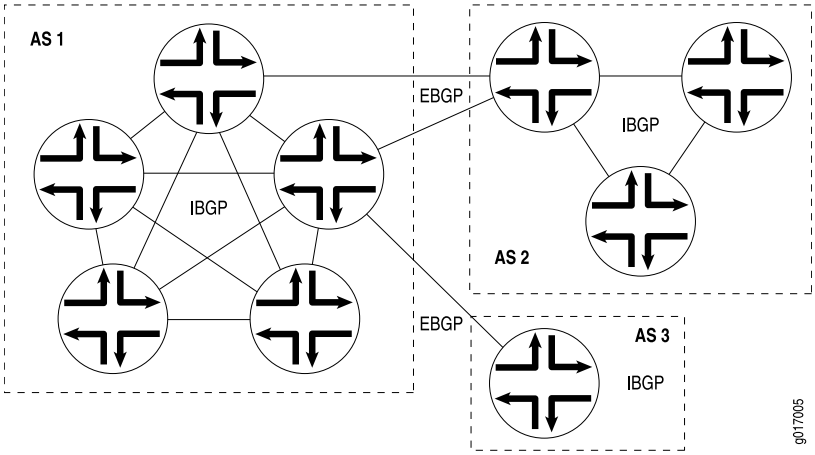
The routing information that BGP systems exchange includes the complete route to each destination, as well as additional information about the route. The route to each destination is called the *AS path*, and the additional route information is included in *path attributes*. BGP uses the AS path and the path attributes to completely determine

the network topology. Once BGP understands the topology, it can detect and eliminate routing loops and select among groups of routes to enforce administrative preferences and routing policy decisions.

External and Internal BGP

BGP supports two types of exchanges of routing information: exchanges between different ASs and exchanges within a single AS. When used between ASs, BGP is called *external BGP* (EBGP) and BGP sessions perform *inter-AS routing*. When used within an AS, BGP is called *internal BGP* (IBGP) and BGP sessions perform *intra-AS routing*. Figure 9 on page 715 illustrates ASs, IBGP, and EBGP.

Figure 9: ASs, EBGP, and IBGP



A BGP system shares network reachability information with adjacent BGP systems, which are referred to as *neighbors* or *peers*.

BGP systems are arranged into *groups*. In an IBGP group, all peers in the group—called *internal peers*—are in the same AS. Internal peers can be anywhere in the local AS and do not have to be directly connected to each other. Internal groups use routes from an IGP to resolve forwarding addresses. They also propagate external routes among all other internal routers running IBGP, computing the next hop by taking the BGP next hop received with the route and resolving it using information from one of the interior gateway protocols.

In an EBGP group, the peers in the group—called *external peers*—are in different ASs and normally share a subnet. In an external group, the next hop is computed with respect to the interface that is shared between the external peer and the local router.

BGP Routes Overview

A BGP route consists of the following:

- A destination, described as an IP address prefix.
- Information that describes the path to the destination, including the following:

- AS path, which is a list of numbers of the ASs that a route passes through to reach the local router. The first number in the path is that of the last AS in the path—the AS closest to the local router. The last number in the path is the AS farthest from the local router, which is generally the origin of the path.
- Path attributes, which contain additional information about the AS path that is used in routing policy.

BGP peers advertise routes to each other in update messages.

BGP stores its routes in the JUNOS Software routing table. The routing table stores the following information about BGP routes:

- Routing information learned from update messages received from peers
- Local routing information that the BGP system selects by applying local policies to routes received in update messages
- Information that the BGP system selects to advertise to its BGP peers in the update messages it sends

For each prefix in the routing table, the routing protocol process selects a single best path, called the active path. The algorithm for determining the active path is described in “How the Active Route Is Determined” on page 7.

Overview of BGP Messages

BGP systems send four types of messages:

- Open
- Update
- Keepalive
- Notification

All BGP messages have the same fixed-size header, which contains a marker field indicating the total length of the message and a type field indicating the message type.

This section discusses the following topics:

- Open Messages on page 717
- Update Messages on page 717
- Keepalive Messages on page 718
- Notification Messages on page 718

Open Messages

After a TCP connection is established between two BGP systems, they exchange BGP open messages to create a BGP connection between them. Once the connection is established, the two systems can exchange BGP messages and data traffic.

Open messages consist of the BGP header plus the following fields:

- Version—The current BGP version number is 4.
- Local AS number—You configure this by including the autonomous-system statement at the [edit routing-options] or [edit logical-systems *logical-system-name* routing-options] hierarchy level, as described in "Specifying the Local Routing Device's AS Number" on page 724.
- Hold time—Proposed hold-time value. You configure the local hold time with the BGP hold-time statement, as described in "Configuring the Delay Before BGP Peers Mark the Routing Device as Down" on page 741.
- BGP identifier—IP address of the BGP system. This address is determined when the system starts up and is the same for every local interface and every BGP peer. You can configure the BGP identifier by including the router-id statement at the [edit routing-options] or [edit logical-systems *logical-system-name* routing-options] hierarchy level, as described in "Assigning a BGP Identifier" on page 725. By default, BGP uses the IP address of the first interface it finds in the router.
- Parameter field length and the parameter itself—These are optional fields.

Update Messages

BGP systems send update messages to exchange network reachability information. BGP systems use this information to construct a graph that describes the relationships among all known ASs.

Update messages consist of the BGP header plus the following optional fields:

- Unfeasible routes length—Length of the field that lists the routes being withdrawn from service because they are no longer deemed reachable
- Withdrawn routes—IP address prefixes for the routes being withdrawn from service
- Total path attribute length—Length of the field that lists the path attributes for a feasible route to a destination
- Path attributes—Properties of the routes, including the path origin, the multiple exit discriminator (MED), the originating system's preference for the route, and information about aggregation, communities, confederations, and route reflection
- Network layer reachability information (NLRI)—IP address prefixes of feasible routes being advertised in the update message

Keepalive Messages

BGP systems exchange keepalive messages to determine whether a link or host has failed or is no longer available. Keepalive messages are exchanged often enough so that the hold timer does not expire. These messages consist only of the BGP header.

Notification Messages

BGP systems send notification messages when an error condition is detected. After the message is sent, the BGP session and the TCP connection between the BGP systems are closed. Notification messages consist of the BGP header plus the error code and subcode, and data that describes the error.

BGP Standards

The JUNOS Software supports BGP version 4 and several extensions to the protocol, which are defined in the following documents:

- RFC 1772, *Application of the Border Gateway Protocol in the Internet*
- RFC 1965, *Autonomous System Confederations for BGP*
- RFC 1966, *BGP Route Reflection: An Alternative to Full-Mesh IBGP*
- RFC 1997, *BGP Communities Attribute*
- RFC 2270, *Using a Dedicated AS for Sites Homed to a Single Provider*
- RFC 2283, *Multiprotocol Extensions for BGP-4*
- RFC 2385, *Protection of BGP Sessions via the TCP MD5 Signature Option*
- RFC 2439, *BGP Route Flap Damping*
- RFC 2545, *Use of BGP-4 Multiprotocol Extensions for IPv6 Inter-Domain Routing*
- RFC 2796, *BGP Route Reflection*
- RFC 2858, *Multiprotocol Extensions for BGP-4*
- RFC 2918, *Route Refresh Capability for BGP-4*
- RFC 3065, *Autonomous System Confederations for BGP*
- RFC 3107, *Carrying Label Information in BGP-4*
- RFC 3392, *Capabilities Advertisement with BGP-4*
- RFC 4271, *A Border Gateway Protocol 4 (BGP-4)*
- RFC 4724, *Graceful Restart Mechanism for BGP*
- RFC 4781, *Graceful Restart Mechanism for BGP with MPLS*
- RFC 4893, *BGP Support for Four-octet AS Number Space*
- Internet draft draft-ietf-ppvpn-rfc2547bis-00.txt, *BGP/MPLS VPNs* (expires January 2002)
- Internet draft draft-kato-bgp-ipv6-link-local-00.txt, *BGP4 + Peering Using IPv6 Link-local Address* (expires April 2002)

- Internet draft draft-ietf-ngtrans-bgp-tunnel-04.txt, *Connecting IPv6 Islands across IPv4 Clouds with BGP* (only MP-BGP over IPv4 Approach) (expires July 2002)
- Internet draft draft-ietf-idr-flow-spec-00.txt, *Dissemination of Flow Specification Rules*

To access Internet RFCs and drafts, go to the Internet Engineering Task Force (IETF) website at http://www.ietf.org .

Chapter 35

BGP Configuration Guidelines

This chapter includes the following topics:

- Configuring BGP on page 722
- Minimum BGP Configuration on page 723
- Enabling BGP on page 724
- Configuring BGP Groups and Peers on page 728
- Examples: Configuring BGP Groups, Peers, and Confederations on page 737
- Configuring the Delay Before BGP Peers Mark the Routing Device as Down on page 741
- Configuring MTU Discovery for BGP Sessions on page 741
- Configuring Graceful Restart for BGP on page 742
- Advertising Explicit Null Labels to BGP Peers on page 742
- Configuring Aggregate Labels for VPNs on page 743
- Configuring Authentication for BGP on page 743
- Using IPsec to Protect BGP Traffic on page 745
- Disabling Transmission of Open Requests to BGP Peers on page 746
- Configuring a Local Endpoint Address for BGP Sessions on page 746
- Configuring the MED in BGP Updates on page 746
- Controlling BGP Route Aggregation on page 750
- Configuring EBGP Multihop Sessions on page 750
- Configuring Single-Hop EBGP Peers to Accept Remote Next Hops on page 750
- Configuring the Local Preference Value for BGP Routes on page 752
- Configuring the Default Preference Value for BGP Routes on page 752
- Configuring Routing Table Path Selection for BGP on page 754
- Selecting Multiple Equal-Cost Active Paths on page 755
- Configuring a Local AS for EBGP Sessions on page 756
- Removing Private AS Numbers from AS Paths on page 761
- Configuring BGP Route Reflection on page 761
- Configuring Flap Damping for BGP Routes on page 767
- Enabling Multiprotocol BGP on page 767

- Enabling BGP to Carry Flow-Specification Routes on page 773
- Enabling BGP to Carry CLNS Routes on page 774
- Enabling BGP Route Target Filtering on page 779
- Applying Filters Provided by BGP Peers to Outbound Routes on page 780
- Enabling Layer 2 VPN and VPLS Signaling on page 781
- Applying Policies to BGP Routes on page 781
- Preventing Automatic Reestablishment of BGP Peering Sessions After NSR Switchovers on page 786
- Configuring EBGP Peering Using IPv6 Link-Local Addresses on page 786
- Configuring IPv6 BGP Routes over IPv4 Transport on page 787
- Configuring System Logging of BGP Peer State Transitions on page 788
- Configuring a Text Description for BGP Groups or Peers on page 788
- Restricting TCP Connections to BGP Peers on page 789
- Applying BGP Export Policy to VRF Routes on page 789
- Including Next-Hop Reachability Information in Multiprotocol Updates on page 789
- Configuring BFD for BGP on page 789
- Overview of BFD Authentication for BGP on page 792
- Configuring BFD Authentication for BGP on page 794
- Limiting TCP Segment Size for BGP on page 798
- Configuring the BGP Monitoring Protocol on page 798
- Tracing BGP Protocol Traffic on page 799

Configuring BGP

To configure BGP, you can include the following statements. Three portions of the bgp statement—those in which you configure global BGP, group-specific, and peer-specific options—contain many of the same statements. The following simplified version of the bgp statement omits these repeated statements to present a high-level, readable overview:

```
protocols {
    bgp {
        ...global-bgp-configuration ...
        group group-name {
            peer-as autonomous-system;
            type type;
            [network/mask-length ];
            ... peer-specific-configuration ...
            neighbor address {
                ... peer-specific-configuration ...
            }
        }
    }
}
```

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for these statements.

For a list of global BGP statements, see “Defining BGP Global Properties” on page 725. For a list of group-specific statements, see “Defining Group Properties” on page 732. For a list of peer-specific statements, see “Defining Peer Properties” on page 734.

NOTE: Changing configuration statements that affect BGP peerings, such as enabling or disabling remove-private or renaming a BGP group, resets the BGP sessions. Changes that affect BGP peerings should only be made when resetting a BGP session is acceptable.

Many of the global BGP, group-specific, and peer-specific statements are identical. For statements that you can configure at more than one level in the hierarchy, the more-specific statement overrides the less-specific statement. That is, a group-specific statement overrides a global BGP statement, and a peer-specific statement overrides a global BGP or group-specific statement.

By default, BGP is disabled.

Minimum BGP Configuration

For BGP to run on the routing device, you must define the local autonomous system (AS) number, configure at least one group, and include information about at least one peer in the group (the peer’s IP address and AS number). There are several ways you can configure this information; a few are shown in this section.

Configure a BGP group, specify the group type, and configure an explicit peer:

```
[edit]
routing-options {
   autonomous-system autonomous-system;
}
protocols {
   bgp {
      group group-name {
         peer-as autonomous-system;
         type type;
         neighbor address;
      }
   }
}
```

Configure a BGP group and type and allow all BGP systems to be peers:

```
[edit]
routing-options {
   autonomous-system autonomous-system;
}
protocols {
   bgp {
      group group-name {
```

```
            type type;
            peer-as autonomous-system;
            all;
        }
    }
}
```

NOTE: When you configure BGP on an interface, you must also include the family inet statement at the [edit interfaces *interface-name* unit *logical-unit-number*] hierarchy level. For more information about the family inet statement, see the *JUNOS Network Interfaces Configuration Guide*.

Enabling BGP

To enable BGP on the routing device, perform the following tasks:

- Specifying the Local Routing Device's AS Number on page 724
- Defining an AS Confederation and Its Members on page 724
- Assigning a BGP Identifier on page 725
- Defining BGP Global Properties on page 725

Specifying the Local Routing Device's AS Number

Each routing device running BGP must be configured with its AS number. This number is included in the local AS number field in BGP open messages, which are sent between BGP peers to establish a connection.

To specify an AS number, include the autonomous-system statement:

```
autonomous-system autonomous-system <loops number>;
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

You must specify an AS number to enable BGP.

For more information about configuring the AS number, see "Configuring AS Numbers for BGP" on page 112.

Defining an AS Confederation and Its Members

To enable the local system to participate as a member of an AS confederation, you must define the AS confederation identifier and specify the AS numbers that are members of the confederation. To do this, include the confederation statement:

```
confederation confederation-autonomous-system members [ autonomous-systems ];
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

Defining an AS confederation and its members is optional.

For more information about configuring confederations, see "Configuring AS Confederation Members" on page 113.

Assigning a BGP Identifier

Each routing device running BGP must have a BGP identifier. This identifier is included in the BGP identifier field of open messages, which are sent between two BGP peers when establishing a BGP session.

Explicitly assigning a BGP identifier is optional. If you do not assign one, the IP address of the first interface encountered in the routing device is used.

To assign a BGP identifier explicitly, include the router-id statement:

```
router-id address;
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

Assigning a BGP identifier is optional.

For more information, see "Configuring Router Identifiers for BGP and OSPF" on page 113.

Defining BGP Global Properties

To define BGP global properties, which apply to all BGP groups and peers, include one or more of the following statements. These statements are explained in separate sections.

```
accept-remote-nexthop;
advertise-external <conditional>;
advertise-inactive;
(advertise-peer-as | no-advertise-peer-as);
authentication-algorithm algorithm;
authentication-key key;
authentication-key-chain key-chain;
bfd-liveness-detection {
    authentication {
        algorithm algorithm-name;
        key-chain key-chain-name;
        loose-check;
    }
    detection-time {
        threshold milliseconds;
    }
    holddown-interval milliseconds;
    minimum-interval milliseconds;
    minimum-receive-interval milliseconds;
    no-adaptation;
    transmit-interval {
        threshold milliseconds;
```

```
        minimum-interval milliseconds;
    }
    multiplier number;
    no-adaptation;
    version (1 | automatic);
}
cluster cluster-identifier;
damping;
description text-description;
disable;
export [ policy-names ];
family {
    (inet | inet6 | inet-vpn | inet6-vpn | iso-vpn) {
        (any | flow | labeled-unicast | multicast | unicast) {
            accepted-prefix-limit {
                maximum number;
                teardown <percentage> <idle-timeout (forever | minutes)>;
            }
            <loops number>;
            prefix-limit {
                maximum number;
                teardown <percentage> <idle-timeout (forever | minutes)>;
            }
            rib-group group-name;
        }
        flow {
            no-validate policy-name;
        }
        labeled-unicast {
            accepted-prefix-limit {
                maximum number;
                teardown <percentage> <idle-timeout (forever | minutes)>;
            }
            aggregate-label {
                community community-name:
            }
            explicit-null {
                connected-only;
            }
            prefix-limit {
                maximum number;
                teardown <percentage> <idle-timeout (forever | minutes)>;
            }
            resolve-vpn;
            rib inet.3;
            rib-group group-name;
        }
    }
    route-target {
        accepted-prefix-limit {
            maximum number;
            teardown <percentage> <idle-timeout (forever | minutes)>;
        }
        advertise-default;
        external-paths number;
        prefix-limit {
```

```
            maximum number;
            teardown <percentage> <idle-timeout (forever | minutes)>;
        }
    }
    (inet-mdt | inet-mvpn | inet6-mvpn | l2-vpn) {
        signaling {
            accepted-prefix-limit {
                maximum number;
                teardown <percentage> <idle-timeout (forever | minutes)>;
            }
            <loops number>;
            prefix-limit {
                maximum number;
                teardown <percentage> <idle-timeout (forever | minutes)>;
                }
            rib-group group-name
        }
    }
}
graceful-restart {
    disable;
    restart-time seconds;
    stale-routes-time seconds;
}
group group-name {
    ... group-specific-options ...
}
hold-time seconds;
idle-after-switch-over (seconds | forever);
import [ policy-names ];
include-mp-next-hop;
ipsec-sa ipsec-sa;
keep (all | none);
local-address address;
local-as autonomous-system <private>;
local-interface interface-name;
local-preference local-preference;
log-updown;
metric-out (metric | minimum-igp <offset> | igp <offset>);
mtu-discovery;
multihop {
    no-nexthop-change;
    ttl value;
}
no-advertise-peer-as;
no-aggregator-id;
no-client-reflect;
out-delay seconds;
outbound-route-filter{
    bgp-orf-cisco-mode;
    prefix-based {
        accept {
            (inet | inet6);
        }
    }
}
```

```
passive;
path-selection {
   (always-compare-med | cisco-non-deterministic | external-router-id);
   med-plus-igp {
      igp-multiplier number;
      med-multiplier number;
   }
}
peer-as autonomous-system;
preference preference;
remove-private;
tcp-mss segment-size;
traceoptions {
   file filename <files number> <size size> <world-readable | no-world-readable>;
   flag flag <flag-modifier> <disable>;
}
vpn-apply-export;
```

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for these statements.

You must configure BGP global properties to enable BGP.

Configuring BGP Groups and Peers

A BGP system must know which routing devices are its peers (neighbors). You define the peer relationships explicitly by configuring the neighboring routers that are the peers of the local BGP system. After peer relationships have been established, the BGP peers exchange update messages to advertise network reachability information.

You arrange BGP routing devices into groups of peers. Different peer groups must have different group types, AS numbers, or route reflector cluster identifiers.

Each group must contain at least one peer.

To configure BGP groups and peers, see the following sections:

- Defining a Group with Static Peers on page 728
- Defining a Group with Dynamic Peers on page 730
- Defining the Group Type on page 731
- Specifying the Peer's AS Number on page 731
- Defining Group Properties on page 732
- Defining Peer Properties on page 734

Defining a Group with Static Peers

To define a BGP group that recognizes only the specified BGP systems as peers, statically configure all the system's peers by including one or more neighbor statements. The peers on at least one side of each BGP connection must be configured statically. The peer neighbor's address can be either an IPv6 or IPv4 address.

```
group group-name {
    peer-as autonomous-system;
    type type;
    neighbor address; # One "neighbor" statement for each peer
}
```

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for these statements.

As the number of EBGP groups increases, the ability to support a large number of BGP sessions may become a scaling issue. The preferred way to configure a large number of BGP neighbors is to configure a few groups consisting of multiple neighbors per group. Supporting fewer EBGP groups generally scales better than supporting a large number of EBGP groups. This becomes more evident in the case of hundreds of EBGP groups when compared with a few EBGP groups with multiple peers in each group. The following examples illustrate this point.

For sample configurations, see the following sections:

- Example: Defining a Large Number of Groups with Static Peers on page 729
- Example: Defining a Small Number of Groups with Static Peers for Better Scalability on page 730

Example: Defining a Large Number of Groups with Static Peers

Enable BGP and define three EBGP groups that recognize BGP systems in AS 56, AS 57, and AS 58 as peers:

```
[edit]
routing-options {
    autonomous-system 23;
}
protocols {
    bgp {
        group G1 {
            type external;
            peer-as 56;
            neighbor 10.0.0.1;
        }
        group G2 {
            type external;
            peer-as 57;
            neighbor 10.0.10.1;
        }
        group G3 {
            type external;
            peer-as 58;
            neighbor 10.0.20.1;
        }
    }
}
```

Example: Defining a Small Number of Groups with Static Peers for Better Scalability

For improved scalability, configure only one EBGP group consisting of the three BGP neighbors:

```
[edit]
routing-options {
   autonomous-system 23;
}
protocols {
   bgp {
      group G {
         type external;
         neighbor 10.0.0.1 {
            peer-as 56;
         }
         neighbor 10.0.10.1 {
            peer-as 57;
         }
         neighbor 10.0.20.1 {
            peer-as 58;
         }
      }
   }
}
```

Defining a Group with Dynamic Peers

To define a BGP group in which the local system's peers are dynamic and change over time, include the allow statement. To recognize all BGP systems as peers, include the allow-all statement. To recognize BGP systems within specified address ranges, specify a set of addresses in the allow *network/mask-length* statement. These addresses can be IPv6 or IPv4 addresses.

```
group group-name {
   peer-as autonomous-system;
   type type;
   allow ([ network/mask-length] | all);
}
```

NOTE: You cannot define a BGP group with dynamic peers with authentication enabled.

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for these statements.

Defining the Group Type

When configuring a BGP group, you can indicate whether the group is an IBGP group or an EBGP group. All peers in an IBGP group are in the same AS, while peers in an EBGP group are in different ASs and normally share a subnet.

To configure an IBGP group, which allows intra-AS BGP routing, include the following form of the type statement:

```
type internal;
```

To configure an EBGP group, which allows inter-AS BGP routing, include the following form of the type statement:

```
type external;
```

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for these statements.

Specifying the Peer's AS Number

When configuring a peer, you must specify the peer system's AS. To do this, include the peer-as statement:

```
peer-as autonomous-system;
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

For *autonomous-system*, you can specify a number of 1 through 4,294,967,295 in plain-number format. In JUNOS Release 9.1 and later, the range for autonomous system (AS) numbers is extended to provide BGP support for 4-byte AS numbers as defined in RFC 4893, *BGP Support for Four-octet AS Number Space*. The JUNOS Software continues to support 2-byte AS numbers.

In JUNOS Release 9.2 and later, you can also configure a 4-byte AS number using the AS-dot notation format of two integer values joined by a period: *<16-bit high-order value in decimal>.<16-bit low-order value in decimal>*. For example, the 4-byte AS number of 65,546 in plain-number format is represented as 1.10 in the AS-dot notation format. You can specify a value in the range from 0.0 through 65535.65535 in AS-dot notation format.

For EBGP, the peer is in another AS, so the AS number you specify in the peer-as statement must be different from the local router's AS number, which you specify in the autonomous-system statement. For IBGP, the peer is in the same AS, so the two AS numbers that you specify in the autonomous-system and peer-as statements must be the same. For more information about configuring the AS number of the local router, see "Configuring AS Numbers for BGP" on page 112.

Defining Group Properties

To define group-specific properties, include one or more of the following statements. For more information, see "Summary of BGP Configuration Statements" on page 803.

```
accept-remote-nexthop;
advertise-external <conditional>;
advertise-inactive;
(advertise-peer-as | no-advertise-peer-as);
allow [ network/mask-length ];
as-override;
authentication-algorithm algorithm;
authentication-key key;
authentication-key-chain key-chain;
bfd-liveness-detection {
    authentication {
        algorithm algorithm-name;
        key-chain key-chain-name;
        loose-check;
    }
    detection-time {
        threshold milliseconds;
    }
    holddown-interval milliseconds;
    minimum-interval milliseconds;
    minimum-receive-interval milliseconds;
    no-adaptation;
    transmit-interval {
        threshold milliseconds;
        minimum-interval milliseconds;
    }
    multiplier number;
    no-adaptation;
    version (1 | automatic);
}
cluster cluster-identifier;
damping;
description text-description;
disable;
export [ policy-names ];
family {
    (inet | inet6 | inet-vpn | inet6-vpn | iso-vpn) {
        (any | flow | labeled-unicast | multicast | unicast) {
            accepted-prefix-limit {
                maximum number;
                teardown <percentage> <idle-timeout (forever | minutes)>;
            }
            <loops number>;
            prefix-limit {
                maximum number;
                teardown <percentage> <idle-timeout (forever | minutes)>;
            }
            rib-group group-name;
```

```
        }
        flow {
            no-validate policy-name;
        }
        labeled-unicast {
            accepted-prefix-limit {
                maximum number;
                teardown <percentage> <idle-timeout (forever | minutes)>;
            }
            aggregate-label {
                community community-name:
            }
            explicit-null {
                connected-only;
            }
            prefix-limit {
                maximum number;
                teardown <percentage> <idle-timeout (forever | minutes)>;
            }
            resolve-vpn;
            rib inet.3;
            rib-group group-name;
        }
    }
    route-target {
        accepted-prefix-limit {
            maximum number;
            teardown <percentage> <idle-timeout (forever | minutes)>;
        }
        advertise-default;
        external-paths number;
        prefix-limit {
            maximum number;
            teardown <percentage> <idle-timeout (forever | minutes)>;
        }
    }
    (inet-mdt | inet-mvpn | inet6-mvpn | l2-vpn) {
        signaling {
            accepted-prefix-limit {
                maximum number;
                teardown <percentage> <idle-timeout (forever | minutes)>;
            }
            <loops number>;
            prefix-limit {
                maximum number;
                teardown <percentage> <idle-timeout (forever | minutes)>;
                }
            rib-group group-name
        }
    }
}
graceful-restart {
    disable;
    restart-time seconds;
    stale-routes-time seconds;
}
```

```
hold-time seconds;
idle-after-switch-over (seconds | forever);
import [ policy-names ];
include-mp-next-hop;
ipsec-sa ipsec-sa;
keep (all | none);
local-address address;
local-as autonomous-system <private>;
local-interface interface-name;
local-preference local-preference;
log-updown;
metric-out (metric | minimum-igp <offset> | igp <offset>);
mtu-discovery;
multihop {
   no-nexthop-change;
   ttl value;
}
multipath {
   multiple-as;
}
neighbor address {
   ... peer-specific-options ...
}
no-aggregator-id;
no-client-reflect;
out-delay seconds;
outbound-route-filter{
   bgp-orf-cisco-mode;
   prefix-based {
      accept {
         (inet | inet6);
      }
   }
}
passive;
peer-as autonomous-system;
preference preference;
remove-private;
tcp-mss segment-size;
traceoptions {
   file filename <files number> <size size> <world-readable | no-world-readable>;
   flag flag <flag-modifier> <disable>;
}
type type;
vpn-apply-export;
```

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for these statements.

Defining Peer Properties

When you use the neighbor statement to configure BGP peers statically, you also can define peer-specific properties. For more information, see “Summary of BGP Configuration Statements” on page 803.

```
accept-remote-nexthop;
advertise-external <conditional>;
advertise-inactive;
(advertise-peer-as | no-advertise-peer-as);
as-override;
authentication-algorithm algorithm;
authentication-key key;
authentication-key-chain key-chain;
bfd-liveness-detection {
   authentication {
      algorithm algorithm-name;
      key-chain key-chain-name;
      loose-check;
   }
   detection-time {
      threshold milliseconds;
   }
   holddown-interval milliseconds;
   minimum-interval milliseconds;
   minimum-receive-interval milliseconds;
   no-adaptation;
   transmit-interval {
      threshold milliseconds;
      minimum-interval milliseconds;
   }
   multiplier number;
   no-adaptation;
   version (1 | automatic);
}
cluster cluster-identifier;
damping;
description text-description;
export [ policy-names ];
family {
   (inet | inet6 | inet-vpn | inet6-vpn | iso-vpn) {
      (any | flow | labeled-unicast | multicast | unicast) {
         accepted-prefix-limit {
            maximum number;
            teardown <percentage> <idle-timeout (forever | minutes)>;
         }
         <loops number>;
         prefix-limit {
            maximum number;
            teardown <percentage> <idle-timeout (forever | minutes)>;
         }
         rib-group group-name;
      }
      flow {
         no-validate policy-name;
      }
      labeled-unicast {
         accepted-prefix-limit {
            maximum number;
            teardown <percentage> <idle-timeout (forever | minutes)>;
         }
         aggregate-label {
```

```
            community community-name;
        }
        explicit-null {
            connected-only;
        }
        prefix-limit {
            maximum number;
            teardown <percentage> <idle-timeout (forever | minutes)>;
        }
        resolve-vpn;
        rib inet.3;
        rib-group group-name;
    }
  }
  route-target {
    accepted-prefix-limit {
        maximum number;
        teardown <percentage> <idle-timeout (forever | minutes)>;
    }
    advertise-default;
    external-paths number;
    prefix-limit {
        maximum number;
        teardown <percentage> <idle-timeout (forever | minutes)>;
    }
  }
  (inet-mdt | inet-mvpn | inet6-mvpn | l2-vpn) {
    signaling {
        accepted-prefix-limit {
            maximum number;
            teardown <percentage> <idle-timeout (forever | minutes)>;
        }
        <loops number>;
        prefix-limit {
            maximum number;
            teardown <percentage> <idle-timeout (forever | minutes)>;
            }
        rib-group group-name
    }
  }
}
graceful-restart {
  disable;
  restart-time seconds;
  stale-routes-time seconds;
}
hold-time seconds;
idle-after-switch-over (seconds | forever);
import [ policy-names ];
include-mp-next-hop;
ipsec-sa ipsec-sa;
keep (all | none);
local-address address;
local-as autonomous-system <private>;
local-interface interface-name;
local-preference preference;
```

```
log-updown;
metric-out (metric | minimum-igp <offset> | igp <offset>);
mtu-discovery;
multihop {
   no-nexthop-change;
   ttl value;
}
multipath {
   multiple-as;
}
no-aggregator-id;
no-client-reflect;
out-delay seconds;
outbound-route-filter{
   bgp-orf-cisco-mode;
   prefix-based {
      accept {
         (inet | inet6);
      }
   }
}
passive;
peer-as autonomous-system;
preference preference;
remove-private;
tcp-mss segment-size;
traceoptions {
   file filename <files number> <size size> <world-readable | no-world-readable>;
   flag flag <flag-modifier> <disable>;
}
vpn-apply-export;
```

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for these statements.

Examples: Configuring BGP Groups, Peers, and Confederations

Enable BGP and define an EBGP group that recognizes all BGP systems in AS 56 as peers:

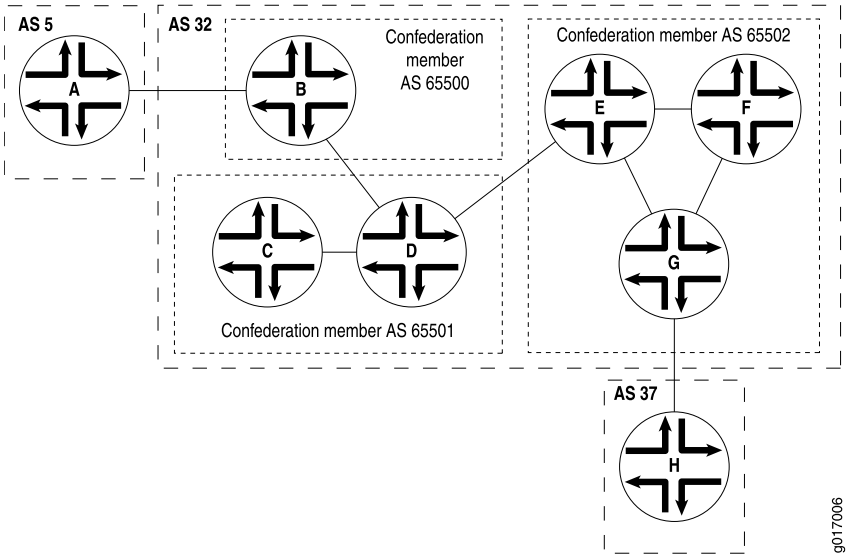
```
[edit]
routing-options {
   autonomous-system 23;
}
protocols {
   bgp {
      group 23 {
         type external;
         peer-as 56;
         0.0.0.0/0;
      }
   }
}
```

Enable BGP and define an IBGP group that recognizes only the specified addresses as BGP peers:

```
[edit]
routing-options {
   autonomous-system 23;
   router-id 10.0.0.1;
}
protocols {
   bgp {
      group 23 {
         type internal;
         peer-as 23;
         neighbor 10.0.0.2;
         neighbor 10.0.0.3;
      }
   }
}
```

Configure a BGP confederation. Figure 10 on page 738 illustrates the confederation topology used in this example. For AS 32 to be a valid confederation, all routers in the AS must be members of the confederation. For example, Router B must have a confederation member AS number as well as a confederation AS number. Within a confederation, the links between the confederation member ASs must be EBGP links, not IBGP links.

Figure 10: Example: BGP Confederation Topology



On Router A:

```
[edit]
routing-options {
   autonomous-system 5;
}
protocols {
   bgp {
```

```
      group AtoB {
        type external;
        peer-as 32;
        neighbor 10.0.0.2;
      }
    }
}
```

On Router B:

```
[edit]
routing-options {
   autonomous-system 65500;
   confederation 32 members [ 65500 65501 65502 ];
}
protocols {
   bgp {
      group BtoA {
         type external;
         peer-as 5;
         neighbor 10.0.0.1;
      }
      group BtoD {
         type external;
         peer-as 65501;
         neighbor 10.0.10.2;
      }
   }
}
```

On Router C:

```
[edit]
routing-options {
   autonomous-system 65501;
   confederation 32 members [ 65500 65501 65502 ];
}
protocols {
   bgp {
      group CtoD {
         type internal;
         neighbor 10.0.10.3;
      }
   }
}
```

On Router D:

```
[edit]
routing-options {
   autonomous-system 65501;
   confederation 32 members [ 65500 65501 65502 ];
}
protocols {
   bgp {
      group DtoC {
         type internal;
         neighbor 10.0.10.1;
      }
      group DtoB {
         type external;
         peer-as 65500;
```

```
        neighbor 10.0.10.1;
      }
      group DtoE {
        type external;
        peer-as 65502;
        neighbor 10.0.30.1;
      }
    }
}
```

On Router E:

```
[edit]
routing-options {
  autonomous-system 65502;
  confederation 32 members [ 65500 65501 65502 ];
}
protocols {
  bgp {
    group EtoD {
      type external;
      peer-as 65501;
      neighbor 10.0.10.4;
    }
    group EtoFandG {
      type internal;
      neighbor 10.0.30.2;
      neighbor 10.0.30.5;
    }
  }
}
```

On Router F:

```
[edit]
routing-options {
  autonomous-system 65502;
  confederation 32 members [ 65500 65501 65502 ];
}
protocols {
  bgp {
    group FtoEandG {
      type internal;
      neighbor 10.0.30.3;
      neighbor 10.0.30.7;
    }
  }
}
```

On Router G:

```
[edit]
routing-options {
  autonomous-system 65502;
  confederation 32 members [ 65500 65501 65502 ];
}
protocols {
  bgp {
    group GtoH {
      type external;
      peer-as 37;
      neighbor 10.0.40.1;
```

```
        }
        group GtoEandF {
            type internal;
            neighbor 10.0.30.4;
            neighbor 10.0.30.5;
        }
    }
}
```

On Router H:

```
[edit]
routing-options {
    autonomous-system 37;
}
protocols {
    bgp {
        group HtoG {
            type external;
            peer-as 32;
            neighbor 10.0.30.8;
        }
    }
}
```

Configuring the Delay Before BGP Peers Mark the Routing Device as Down

The hold time is the maximum number of seconds allowed to elapse between successive keepalive or update messages that BGP receives from a peer. When establishing a BGP connection with the local routing device, a peer sends an open message, which contains a hold-time value. BGP on the local routing device uses the smaller of either the local hold-time value or the peer's hold-time value received in the open message as the hold time for the BGP connection between the two peers.

To modify the hold-time value on the local BGP system, include the hold-time statement:

```
hold-time seconds;
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

The default hold-time value is 90 seconds.

The range is 20 through 65,535 seconds.

The hold time is three times the interval at which keepalive messages are sent.

Configuring MTU Discovery for BGP Sessions

You can configure TCP path maximum transmission unit (MTU) discovery. MTU discovery improves convergence times for IBGP sessions. BGP unconditionally disables TCP path MTU discovery, resulting in a 512-byte MSS on TCP sessions that are not directly connected. This feature allows you to to enable TCP path MTU discovery on BGP sessions.

To configure MTU discovery, include the mtu-discovery statement:

```
mtu-discovery;
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

Configuring Graceful Restart for BGP

Graceful restart is disabled by default. You can globally enable graceful restart for all routing protocols at the [edit routing-options] or [edit logical-systems *logical-system-name* routing-options] hierarchy levels.

To configure graceful restart specifically for BGP, include the graceful-restart statement:

```
graceful-restart {
    restart-time;
    stale-routes-time;
}
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

NOTE: Configuring graceful restart for BGP resets the BGP peer routing statistics to zero.

To disable graceful restart for BGP, specify the disable statement. To configure a time period to complete restart, specify the restart-time statement. To configure a time period over which to keep stale routes during a restart, specify the stale-routes-time statement.

Advertising Explicit Null Labels to BGP Peers

You can advertise an explicit null label (label 0) out of the egress router for a label-switched path (LSP). By default, the routing device advertises label 3. Enabling explicit null allows the routing device to send out label 0. Advertising explicit null labels is used for peers in the same BGP group.

Configure the labeled-unicast statement with the explicit-null option. As with regular BGP configuration, the family statement can be specified.

To advertise an explicit null label, include the following statements in the configuration:

```
family inet {
    labeled-unicast {
        aggregate-label {
            community community-name:
        }
        explicit-null {
            connected-only;
```

```
        }
    }
}
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

Specify the connected-only statement to advertise explicit null labels between connected routes only (direct routes and loopback routes).

NOTE: The connected-only statement is required to advertise explicit null labels.

NOTE: Explicit null labels are supported for the IPv4 (inet) family only.

Configuring Aggregate Labels for VPNs

Aggregate labels for VPNs allow a Juniper Networks routing device to aggregate a set of incoming labels (labels received from a peer routing device) into a single forwarding label that is selected from the set of incoming labels. The single forwarding label corresponds to a single next hop for that set of labels.

For a set of labels to share a single forwarding label, they must belong to the same forwarding equivalence class (FEC). The labeled packets must have the same destination egress interface.

To configure aggregate labels for VPNs, include the aggregate-label statement:

```
aggregate-label {
    community community-name;
}
```

For a list of hierarchy levels at which you can include the aggregate-label statement, see the statement summary for this statement.

Configuring Authentication for BGP

All BGP protocol exchanges can be authenticated to guarantee that only trusted routing devices participate in the AS's routing. By default, authentication is disabled. You can configure MD5 authentication. The MD5 algorithm creates an encoded checksum that is included in the transmitted packet. The receiving routing device uses an authentication key (password) to verify the packet's MD5 checksum.

To configure an MD5 authentication key, include the authentication-key statement:

```
authentication-key key;
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

If you configure authentication for all peers, each individual peer in that group inherits the group's authentication.

The key (password) can be up to 126 characters long. Characters can include any ASCII strings. If you include spaces, enclose all characters in quotation marks (" ").

You can update MD5 authentication keys without resetting any BGP peering sessions. This is referred to as hitless authentication key rollover. Hitless authentication key rollover uses authentication keychains, which consist of the authentication keys that are being updated.

Hitless authentication key rollover also allows users to choose the algorithm through which authentication is established. The user associates a keychain and an authentication algorithm with a BGP neighboring session. The keychain includes multiple keys. Each key contains an identifier and a secret. The key is also configured with a unique start time and an end time.

The sending peer chooses the active key based on the system time. The receiving peer determines the key with which it authenticates based upon the incoming key identifier.

To configure the authentication key, include the key-chain statement at the [edit security authentication-key-chains] hierarchy level, and specify the key option to create a keychain consisting of several authentication keys.

```
[edit security]
authentication-key-chains {
    key-chain key-chain-name {
        key key {
            secret secret-data;
            start-time YYYY-MM-DD.hh:mm:ss;
        }
    }
}
```

You can configure multiple keys within the keychain.

Each key within a keychain must be identified by a unique integer value configured in the key statement. The range of valid identifier values is from 0 through 63. Each key must specify a secret. This secret can be entered in either encrypted or plain text format in the secret statement. It is always displayed in encrypted format.

Each key must specify a start time with the start-time statement. Start times are specified in the local time zone for a routing device and must be unique within the key chain.

For more information on configuring authentication keychains, see the *JUNOS System Basics Configuration Guide*.

To apply an authentication keychain, include the authentication-key-chain statement:

```
authentication-key-chain key-chain;
```

To specify the authentication algorithm type to use for keychains, include the authentication-algorithm statement:

```
authentication-algorithm algorithm;
```

You can choose either md5 or hmac-sha-1-96 as the type of algorithm.

NOTE: BGP authentication is not supported with promiscuous mode BGP sessions. If you include the allow statement, you cannot include authentication-key or authentication-key-chain at the same hierarchy level or any higher hierarchy level. When configuring authentication for all peers in a group, you cannot include the allow statement in the configuration because BGP keys require a destination address.

For a list of hierarchy levels at which you can include the previous statements, see the statement summary for those statements.

Using IPsec to Protect BGP Traffic

You can apply IPsec to BGP traffic. IPsec is a protocol suite used for protecting IP traffic at the packet level. IPsec is based on security associations (SAs). An SA is a simplex connection that provides security services to the packets carried by the SA. After configuring the SA, you can apply it to BGP peers.

To apply an SA, include the ipsec-sa statement:

```
ipsec-sa ipsec-sa;
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement. The security association is identified by the SA name.

NOTE: For transport mode, no PIC is necessary. The SA is configured at the [edit security ipsec security-association name] hierarchy level with the mode statement set to transport. In transport mode, the JUNOS Software does not support authentication header (AH) or encapsulating security payload (ESP) header bundles. The JUNOS Software supports only the BGP protocol in transport mode.

For tunnel mode, a MultiServices PIC (or MS-DPC for MX eries routers) must be used. Tunnel mode IPSec for MS-PIC is configured at the [edit security ipsec-vpn] hierarchy level.

A more specific SA overrides a less general SA. For example, if a specific SA is applied to a specific peer, that SA overrides the SA applied to the whole peer group.

For more detailed information about configuring IPsec security associations, see the *JUNOS System Basics Configuration Guide*.

Disabling Transmission of Open Requests to BGP Peers

You can configure a routing device not to send Open requests to a peer. Once you configure the routing device to be passive, the routing device does not originate the TCP connection. However, when the routing device receives a connection from the peer and an Open message, it replies with another BGP Open message. Each routing device declares its own capabilities.

To configure the routing device so that it does not send Open requests to a peer, include the passive statement:

```
passive;
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

Configuring a Local Endpoint Address for BGP Sessions

You can specify the address of the local end of a BGP session. You generally do this to explicitly configure the system's IP address from BGP's point of view. This IP address can be either an IPv6 or IPv4 address. Typically, an IP address is assigned to a loopback interface, and that IP address is configured here. This address is used to accept incoming connections to the peer and to establish connections to the remote peer. To assign a local address, include the local-address statement:

```
local-address address;
```

NOTE: A BGP session can still be established when only one of the paired routers has a local address configured.

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

If you include the default-address-selection statement in the configuration, the software chooses the system default address as the source for most locally generated IP packets. For more information, see the *JUNOS System Basics Configuration Guide*. For protocols in which the local address is unconstrained by the protocol specification, for example IBGP and multihop EBGP, if you do not configure a specific local address when configuring the protocol, the local address is chosen using the same methods as other locally generated IP packets.

Configuring the MED in BGP Updates

The BGP multiple exit discriminator (MED, or MULTI_EXIT_DISC) is an optional path attribute that can be included in BGP update messages. This attribute is used on external BGP links (that is, on inter-AS links) to select among multiple exit points to a neighboring AS. The MED attribute has a value that is referred to as a *metric*. If all

other factors in determining an exit point are equal, the exit point with the lowest metric is preferred.

If a MED is received over an external BGP link, it is propagated over internal links to other BGP systems within the AS.

BGP update messages include a MED metric if the route was learned from BGP and already had a MED metric associated with it, or if you configure the MED metric in the configuration file in one of the following ways:

- Defining a MED Metric Directly on page 747
- Using Routing Policy to Define a MED Metric on page 748
- Examples: Configuring the MED Metric on page 748

A MED metric is advertised with a route according to the following general rules:

- A more specific metric overrides a less specific metric. That is, a group-specific metric overrides a global BGP metric and a peer-specific metric overrides a global BGP or group-specific metric.
- A metric defined with routing policy overrides a metric defined with the metric-out statement.
- If any metric is defined, it overrides a metric received in a route.
- If the received route does not have an associated MED metric, and if you do not explicitly configure a metric value, no metric is advertised. When you do not explicitly configure a metric value, the MED is equivalent to zero (0) when advertising an active route.

For a description of the algorithm used to determine the active path, see "How the Active Route Is Determined" on page 7.

Defining a MED Metric Directly

To directly configure a MED metric to advertise in BGP update messages, include the metric-out statement:

metric-out (*metric* | minimum-igp *offset* | igp delay-med-update | *offset*);

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

metric is the primary metric on all routes sent to peers. It can be a value in the range from 0 through 4,294,967,295 ($2^{32} - 1$).

Specify minimum-igp to set the metric to the minimum metric value calculated in the IGP to get to the BGP next hop. If a newly calculated metric is greater than the minimum metric value, the metric value remains unchanged. If a newly calculated metric is lower, the metric value is lowered to that value.

Specify igp to set the metric to the most recent metric value calculated in the IGP to get to the BGP next hop.

In JUNOS Release 9.0 and later, you can also specify that a BGP group or peer configured with the metric-out igp statement delay sending MED updates when the MED value increases. Include the delay-med-update statement when you configure the igp statement. The default interval to delay sending updates unless the MED is lower or another attribute associated with the route has changed is 10 minutes. Include the med-igp-update-interval *minutes* statement at the [edit routing-options] hierarchy level to modify the default interval. For information, see "Delaying Updates of the MED Path Attribute for BGP" on page 133.

Specify a value for *offset* to increase or decrease the metric that is used from the metric value calculated in the IGP. The metric value is offset by the value specified. The metric calculated in the IGP (by specifying either igp or igp-minimum) is increased if the *offset* value is positive. The metric calculated in the IGP (by specifying either igp or igp-minimum) is decreased if the *offset* value is negative.

offset can be a value in the range from -2^{31} through $2^{31} - 1$. Note that the adjusted metric can never go below 0 or above $2^{32} - 1$.

Using Routing Policy to Define a MED Metric

To use routing policy to define a MED metric to advertise, define the routing policy by including the policy-statement statement at the [edit policy-options] hierarchy level, and then apply the filter by including the import and export statements when configuring BGP.

When defining the routing policy filter, include an action that specifies the desired metric value:

```
[edit policy-options]
policy-statement policy-name {
   term term-name {
      from {
         match-conditions;
         prefix-list name;
         route-filter destination-prefix match-type <actions>;
      }
      to {
         match-conditions;
      }
      then actions;
   }
}
```

For information about defining routing policy, see the *JUNOS Policy Framework Configuration Guide*. For information about applying filters in BGP, see "Applying Policies to BGP Routes" on page 781.

Examples: Configuring the MED Metric

Set the MED metric to 20 for all routes advertised in BGP update messages except for those sent to the peer system 192.168.0.1; the MED for this peer is 10:

```
[edit]
```

```
routing-options {
   router-id 10.0.0.1;
   autonomous-system 23;
}
protocols {
   bgp {
      metric-out 20;
      group 23 {
         type external;
         peer-as 56;
         neighbor 192.168.0.1 {
            traceoptions {
               file bgp-log-peer;
               flag packets;
            }
            log-updown;
            metric-out 10;
         }
      }
   }
}
```

Set the MED metric to 20 for all routes from a particular community:

```
[edit]
routing-options {
   router-id 10.0.0.1;
   autonomous-system 23;
}
policy-options {
   policy-statement from-otago {
      from community otago;
      then metric 20;
   }
   community otago members [56:2379 23:46944];
}
protocols {
   bgp {
      import from-otago;
      group 23 {
         type external;
         peer-as 56;
         neighbor 192.168.0.1 {
            traceoptions {
               file bgp-log-peer;
               flag packets;
            }
            log-updown;
         }
      }
   }
}
```

Controlling BGP Route Aggregation

The JUNOS Software implementation of BGP performs route aggregation, which is the process of combining the characteristics of different routes so that only a single route is advertised. Aggregation reduces the amount of information that BGP must store and exchange with other BGP systems.

BGP adds the aggregator path attribute to BGP update messages. This attribute contains the local system's AS number and IP address (router ID).

To prevent different routers within an AS from creating aggregate routes that contain different AS paths, set the IP address in the aggregator path attribute to 0 by including the no-aggregator-id statement:

```
no-aggregator-id;
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

Configuring EBGP Multihop Sessions

If an EBGP peer is more than one hop away from the local router, you must specify the next hop to the peer so that the two systems can establish a BGP session. This type of session is called a *multihop* BGP session. To configure a multihop session, include the multihop statement:

```
multihop {
    <ttl-value>;
    no-nexthop-change;
}
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

To configure the maximum time-to-live (TTL) value for the TTL in the IP header of BGP packets, specify *ttl-value*. If you do not specify a TTL value, the system's default maximum TTL value is used. To specify not to change the BGP next-hop value for route advertisements, specify the no-nexthop-change option.

Configuring Single-Hop EBGP Peers to Accept Remote Next Hops

In some situations, it is necessary to configure a single-hop EBGP peer to accept a remote next hop with which it does not share a common subnet. The default behavior is for any next-hop address received from a single-hop EBGP peer that is not recognized as sharing a common subnet to be discarded. The ability to have a single-hop EBGP peer accept a remote next hop to which it is not directly connected also prevents you from having to configure the single-hop EBGP neighbor as a multihop session. When you configure a multihop session in this situation, all next-hop routes learned through this EBGP peer are labeled indirect even when they do share a common subnet. This situation breaks multipath functionality for routes that are

recursively resolved over routes that include these next-hop addresses. Configuring the accept-remote-nexthop statement allows next-hop routes that share a common subnet to be installed as direct, which restores multipath functionality for routes that are resolved over these next-hop addresses. Both the remote next-hop and the EBGP peer must support BGP route refresh as defined in RFC 2918, *Route Refresh Capability in BGP-4*. If the remote peer does not support BGP route refresh, the session is reset.

NOTE: You cannot configure both the multihop and accept-remote-nexthop statements for the same EBGP peer.

When you enable a single-hop EBGP peer to accept a remote next hop, you must also configure an import routing policy on the EBGP peer that specifies the remote next-hop address. For more information about how to configure a BGP routing policy, see "Applying Policies to BGP Routes" on page 781 and the *JUNOS Policy Framework Configuration Guide*.

To enable a single-hop EBGP peer to accept a remote next hop, include the accept-remote-nexthop statement:

```
accept-remote-nexthop;
```

You can configure this statement at the global, group, and neighbor hierarchy levels for BGP. The statement is also supported on logical systems and the VPN routing and forwarding (VRF) routing instance type. For a complete list of hierarchy levels at which you can configure this statement, see the statement summary section for this statement.

Example: Configure an Import Routing Policy for an EBGP Peer to Accept a Remote Next Hop

Configure an import routing policy, agg_route, that enables a single-hop external BGP peer to accept the remote next-hop 1.1.10.10. At the [edit protocols bgp] hierarchy level, include the import agg_route statement to apply the policy to the external BGP peer and include the accept-remote-nexthop statement to enable the single-hop EBGP peer to accept the remote next hop.

```
[edit]
policy-options {
   policy-statement agg_route {
      term 1 {
         from {
            protocol bgp;
            route-filter 1.1.230.0/23 exact;
         }
         then {
            next-hop 1.1.10.10;
            accept;
         }
      }
   }
}
protocols {
```

```
    bgp {
        accept-remote-nexthop;
        group ext {
            type external;
            import agg_route;
            peer-as 65001;
            multipath;
            neighbor 1.1.0.1;
            neighbor 1.1.1.1;
        }
        group int {
            type internal;
            local-address 10.255.71.24;
            neighbor 10.255.14.177;
        }
    }
}
```

Configuring the Local Preference Value for BGP Routes

IBGP sessions use a metric called the *local preference*, which is carried in IBGP update packets in the path attribute LOCAL_PREF. This metric indicates the degree of preference for an external route. The route with the highest local preference value is preferred.

The LOCAL_PREF path attribute is always advertised to internal BGP peers and to neighboring confederations. It is never advertised to external BGP peers. The default behavior is to not modify the LOCAL_PREF path attribute if it is present.

NOTE: The LOCAL_PREF path attribute applies at export time only.

By default, if a received route contains a LOCAL_PREF path attribute value, the value is not modified. If a BGP route is received without a LOCAL_PREF attribute, the route is handled locally (that is, it is stored in the routing table and advertised by BGP) as if it were received with a LOCAL_PREF value of 100. A non-BGP route that is advertised by BGP is advertised with a LOCAL_PREF value of 100 by default.

To change the local preference metric advertised in the path attribute, include the local-preference statement, specifying a value from 0 through 4,294,967,295 ($2^{32} - 1$):

local-preference *local-preference*;

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

Configuring the Default Preference Value for BGP Routes

When the JUNOS Software determines a route's preference to become the active route, it selects the route with the lowest preference as the active route and installs this route into the forwarding table. By default, the routing software assigns a preference of 170 to routes that originated from BGP. Of all the routing protocols,

BGP has the highest default preference value, which means that routes learned by BGP are the least likely to become the active route. (For more information about preferences, see “Route Preferences Overview” on page 6.)

To modify the default BGP preference value, include the **preference** statement, specifying a value from 0 through 4,294,967,295 ($2^{32} - 1$):

```
preference preference;
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

Examples: Configuring BGP Route Preference

Assign a preference of **160** to routes learned from the BGP system **192.168.1.1**. The routing protocol process prefers these routes over routes learned from other BGP systems, which have the default preference of 170.

```
[edit]
routing-options {
    autonomous-system 23;
}
protocols {
    bgp {
        group 23 {
            type external;
            peer-as 56;
            neighbor 192.168.1.1 {
                preference 160;
            }
        }
    }
}
```

Assign a preference of **140** to all routes learned by BGP systems. Because the default OSPF preference is 150, BGP routes are preferred over those learned from OSPF.

```
[edit]
routing-options {
    autonomous-system 23;
}
protocols {
    bgp {
        preference 140;
        group 23 {
            type external;
            peer-as 56;
            neighbor 192.168.1.1;
        }
    }
}
```

Configuring Routing Table Path Selection for BGP

By default, only the multiple exit discriminators (MEDs) of routes that have the same peer autonomous systems (ASs) are compared. You can configure routing table path selection options to get different behaviors. To configure routing table path selection behavior, include the path-selection statement:

```
path-selection {
   (cisco-non-deterministic | always-compare-med | external-router-id);
   med-plus-igp {
      igp-multiplier number;
      med-multiplier number;
   }
}
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

Routing table path selection can be configured in one of the following ways:

- Using the same nondeterministic behavior as does the Cisco IOS software (cisco-non-deterministic). This behavior has two effects:
  - The active path is always first. All nonactive, but eligible, paths follow the active path and are maintained in the order in which they were received, with the most recent path first. Ineligible paths remain at the end of the list.
  - When a new path is added to the routing table, path comparisons are made without removing from consideration those paths that should never be selected because those paths lose the MED tie-breaking rule.

These two effects cause the system to only sometimes compare the MEDs between paths that it should otherwise compare. Because of this, we recommend that you not configure nondeterministic behavior.

- Always comparing MEDs whether or not the peer ASs of the compared routes are the same (always-compare-med).

For an example of always comparing MEDs, see "Example: Always Comparing MEDs" on page 755.

- Comparing the router ID between external BGP paths to determine the active path (external-router-id). By default, router ID comparison is not performed if one of the external paths is active.
- Adding the IGP cost to the next-hop destination to the MED before comparing MED values for path selection.

For a description of the algorithm used to determine the active path, see "How the Active Route Is Determined" on page 7.

Example: Always Comparing MEDs

In this example, paths learned from 208.197.169.15 have their MED values compared to the sum of 4 and the MED values of the same paths learned from 208.197.169.14:

```
[edit]
protocols {
    bgp {
        path-selection always-compare-med;
        group ref {
            type external;
            import math;
            peer-as 10458;
            neighbor 208.197.169.14;
        }
        group ref {
            type external;
            peer-as 10;
            neighbor 208.197.169.15;
        }
    }
}
policy-options {
    policy-statement math {
        then {
            metric add 4;
        }
    }
}
```

Selecting Multiple Equal-Cost Active Paths

You can configure BGP to select multiple equal-cost EBGP or IBGP paths as active paths. Selecting multiple paths allows BGP peerings to load-balance traffic across an AS-confederation boundary. The JUNOS Software BGP multipath supports the following:

- Load balancing across multiple links between two routing devices belonging to different ASs
- Load balancing across a common subnet or multiple subnets to different routing devices belonging to the same peer AS
- Load balancing across multiple links between two routing devices belonging to different external confederation peers
- Load balancing across a common subnet or multiple subnets to different routing devices belonging to external confederation peers

To configure a BGP multipath, include the multipath statement:

```
multipath {
    multiple-as;
}
```

To disable the default check requiring that paths accepted by BGP multipath must have the same neighboring AS, include the multiple-as option.

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

Configuring a Local AS for EBGP Sessions

When an Internet service provider (ISP) acquires a network that belongs to a different autonomous system (AS), there is no seamless method for moving the BGP peers in the acquired network to the AS of the acquiring ISP.. The process of configuring the acquired BGP peers with the new AS number can be time-consuming and cumbersome. Moreover, the customers of the acquired network either might not want to or immediately be able to modify their peering arrangements or configuration. During such a transition period, it can be useful to configure BGP peers that have been migrated to the new AS to also use the former AS number in BGP updates. This second AS is called a *local AS*. The use of a local AS number permits the routing devices in an acquired network to appear to belong to two ASs: the new AS (the global AS) to which it now physically belongs and the former AS. All routing devices running BGP must be configured with a global AS by including the autonomous-system *number* statement at the [edit routing-options] hierarchy level.

The JUNOS Software implementation of the local AS feature supports the following options:

- **Local AS**—Configure a local AS for a BGP peer so that both the global AS and the local AS are used in BGP inbound and outbound updates. The local AS is prepended before the global AS in the AS path used by the BGP peer sent both to internal (IBGP) neighbors and external (EBGP) peers.
- **Local AS with private option**—When you use the private option, the local AS is used during the establishment of the BGP session with an external (EBGP) neighbor but is hidden in the AS path sent to other EBGP peers. Only the global AS is included in the AS path sent to other external peers.
- **Local AS with alias option**—In JUNOS Release 9.5 and later, you can configure a local AS as an alias. During the establishment of the BGP open session, the AS used in the Open message alternates between the local AS and the global AS. If the local AS is used to connect with the EBGP neighbor, then only the local AS is prepended to the AS path when the BGP peering session is established. If the global AS is used to connect with the EBGP neighbor then only the global AS is prepended to the AS path when the BGP peering session is established. The use of the alias option also means that the local AS is not prepended to the AS path for any routes learned from that EBGP neighbor. Therefore, the local AS remains hidden from other external peers.
- **Local AS with option not to prepend the global AS**—In JUNOS Release 9.6 and later, you can configure a local AS with the option not to prepend the global AS. Only the local AS is included in the AS path sent to external peers.
- **Number of loops option**—The local AS feature also supports the ability to specify the maximum number of times the AS can appear in the AS path, to prevent routing loops.

Configuring a local AS that is used in inbound and outbound BGP updates is particularly useful when the customer of an acquired network provider does not want to or is not immediately able to modify its peering arrangements or configuration. For example, ISP A, with an AS of 1000, acquires ISP B, with an AS of 100. ISP B's customer, ISP C, does not want to change its configuration. After ISP B becomes part of ISP A, a local AS number of 100 is configured for use in EBGP peering sessions with ISP C. This means that the local AS value of 100 is prepended before the global AS value of 1000 in the AS path used to export routes to direct external peers in ISP C.

Configuring a local AS with the alias option is especially useful when you are migrating the routing devices in an acquired network to the new AS. During the migration process, some routing devices might be configured with the new AS while others remain configured with the former AS. For example, it is good practice to start by migrating first to the new AS any routing devices that function as route reflectors. However, as you migrate the route reflector clients incrementally, the route reflector has to peer with routing devices configured with the former AS as well as routing devices configured with the new AS. To establish local peering sessions, it can be useful for the BGP peers in the network to be able to use both the local AS and the global AS. At the same time, you want to hide this local AS from external peers and use only the global AS in the AS path when exporting routes to another AS. In such situations, choose the alias option.

The private option is useful for establishing local peering with routing devices that remain configured with their former AS or with a specific customer that has not yet modified its peering arrangements. The local AS is used to establish the BGP session with the EBGP neighbor but is hidden in the AS path sent to external peers in another AS.

Use the no-prepend-global-as option when you want to strip the global AS number from outbound BGP updates. This option is useful in a virtual private network (VPN) scenario where you want to hide the global AS from the VPN.

NOTE: If the local AS for the EBGP or IBGP peer is the same as the current AS, do not use the local-as statement to specify the local AS number.

To configure a local AS, include the local-as statement:

local-as *autonomous-system* <loops *number*> <private | alias> no-prepend-global-as;

The local-as statement is supported for BGP at the global, group, and neighbor hierarchy levels.

For *autonomous-system*, you can specify a number from 1 through 4,294,967,295 in plain-number format. In JUNOS Release 9.1 and later, the range for autonomous system (AS) numbers is extended to provide BGP support for 4-byte AS numbers as defined in RFC 4893, *BGP Support for Four-octet AS Number Space*. The JUNOS Software continues to support 2-byte AS numbers.

In JUNOS Release 9.3 and later, you can also configure a 4-byte AS number using the AS-dot notation format of two integer values joined by a period: < *16-bit high-order value in decimal* > . < *16-bit low-order value in decimal* > . For example, the 4-byte AS

number of 65,546 in plain-number format is represented as 1.10 in the AS-dot notation format. You can specify a value from 0.0 through 65535.65535 in AS-dot notation format.

Include the private option so that the local AS is not prepended before the global AS in the AS path sent to external peers. When you specify the private option, the local AS is prepended only in the AS path sent to the EBGP neighbor.

Include the alias option to configure the local AS as an alias to the global AS configured at the [edit routing-options] hierarchy level. When you configure a local AS as an alias, during the establishment of the BGP open session, the AS used in the Open message alternates between the local AS and the global AS. The local AS is prepended to the AS path only when the peering session with an EBGP neighbor is established using that local AS. The local AS is hidden in the AS path sent to any other external peers. Only the global AS is prepended to the AS Path when the BGP session is established using the global AS.

NOTE: The private and alias options are mutually exclusive. You cannot configure both options with the same local-as statement.

Include the no-prepend-global-as option to have the global AS configured at the [edit routing-options] hierarchy level stripped from the AS path sent to external peers. When you use this option, only the local AS is included in the AS path.

Include the loops *number* statement to specify the maximum number of times the local AS can appear in an AS Path. For *number*, specify a value from 1 through 10.

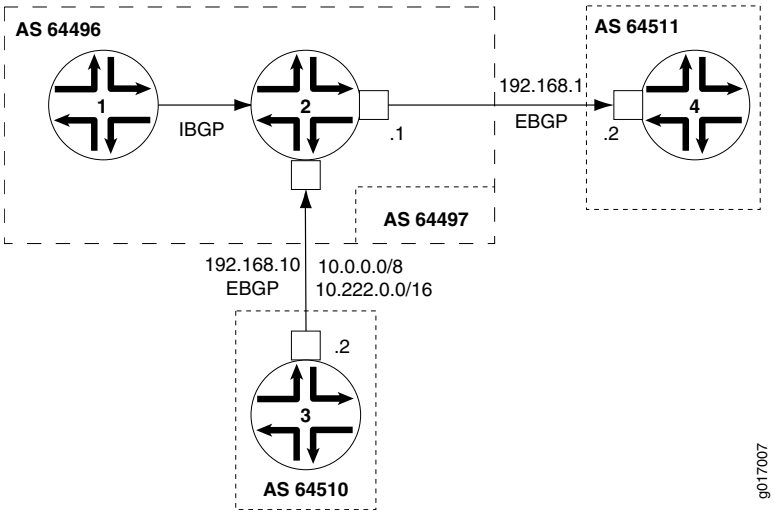
Examples: Configuring a Local AS

You can include the local-as statement to configure a router to use a different AS number than the one for which the router is configured. The local AS is used in all BGP protocol exchanges with the routers that are configured for simulating a virtual AS.

NOTE: If you configure the local AS values for any BGP group, the detection of routing loops is performed using both the AS and the local AS values for all BGP groups.

Use the local-as statement when ISPs merge and want to preserve a customer's configuration, particularly the AS with which the customer is configured to establish a peering relationship. Use the local-as statement to simulate the AS number already in place in customer routers, even if the ISP's router has moved to a different AS.

Figure 11: Local AS Configuration



In Figure 11 on page 759, Router 1 and Router 2 are in AS 64496, Router 4 is in AS 64511, and Router 3 is in AS 64510. Router 2 used to belong to AS 64497, which has merged with another network and now belongs to AS 64496. Because Router 3 still peers with Router 2 using its former AS, 64497, Router 2 needs to be configured with a local AS of 64497 to maintain peering with Router 3. Configuring a local AS of 64497 permits Router 2 to add AS 64497 when advertising routes to Router 3. Router 3 sees an AS path of 64497 64496 for the prefix 10/8.

To prevent Router 2 from adding the local AS number in its announcements to other peers, use the local-as 64497 private statement. This statement configures Router 2 to not include the local AS 64497 when announcing routes to Router 1 and to Router 4. In this case, Router 4 sees an AS path of 64496 64510 for the prefix 10.222/16.

The configuration for each router follows.

On Router 1:

```
routing-options {
   autonomous-system 64496;
}
protocols {
   bgp {
      group internal-AS64496 {
         type internal;
         local-address 10.1.1.1;
         neighbor 10.1.1.2;
      }
   }
}
```

On Router 2:

```
routing-options {
   autonomous-system 64496;
}
protocols {
   bgp {
```

```
        group internal-AS64496 {
            type internal;
            local-address 10.1.1.2;
            neighbor 10.1.1.1;
        }
        group external-AS64511 {
            type external;
            peer-as 64511;
            neighbor 192.168.1.2;
        }
        group external-AS64510 {
            type external;
            peer-as 64511;
            local-as 64497 private;
            neighbor 192.168.10.2;
        }
    }
}
```

On Router 3:

```
routing-options {
    autonomous-system 64510;
}
protocols {
    bgp {
        group external-AS64497 {
            type external;
            peer-as 64497;
            neighbor 192.168.10.1;
        }
    }
}
```

On Router 4:

```
routing-options {
    autonomous-system 64511;
}
protocols {
    bgp {
        group externl-64496 {
            peer-as 64496;
            neighbor 192.168.1.1;
        }
    }
}
```

Related Topics ■ autonomous-system

Removing Private AS Numbers from AS Paths

By default, when BGP advertises AS paths to remote systems, it includes all AS numbers, including private AS numbers. You can configure the software so that it removes private AS numbers from AS paths. Doing this is useful when all the following circumstances are true:

- A remote AS for which you provide connectivity is multihomed, but only to the local AS.
- The remote AS does not have an officially allocated AS number.
- It is not appropriate to make the remote AS a confederation member AS of the local AS.

To have the local system strip private AS numbers from the AS path, include the remove-private statement:

```
remove-private;
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

CAUTION: Changing configuration statements that affect BGP peerings, such as enabling or disabling remove-private or renaming a BGP group, resets the BGP sessions. Changes that affect BGP peerings should only be made when resetting a BGP session is acceptable.

NOTE: The remote-private statement is applicable only when advertising routers to another neighbor.

The AS numbers are stripped from the AS path starting at the left end of the AS path (the end where AS paths have been most recently added). This operation takes place after any confederation member ASs have already been removed from the AS path, if applicable.

The software is preconfigured with knowledge of the set of AS numbers that is considered private, a range that is defined in the Internet Assigned Numbers Authority (IANA) assigned numbers document. The set of AS numbers reserved as private are in the range from 64,512 through 65,534, inclusive.

Configuring BGP Route Reflection

In standard IBGP implementations, all BGP systems within the AS are fully meshed so that any external routing information is redistributed among all routing devices within the AS. This type of implementation can present scaling issues when an AS

has a large number of internal BGP systems because of the amount of identical information that BGP systems must share with each other. Route reflection provides one means of decreasing BGP control traffic and minimizing the number of update messages sent within the AS.

In route reflection, BGP systems are arranged in *clusters*. Each cluster consists of at least one system that acts as a *route reflector*, along with any number of *client peers*. BGP peers outside the cluster are called *nonclient peers*. The route reflector reflects (redistributes) routing information to each client peer (*intracluster reflection*) and to all nonclient peers (*intercluster reflection*). Because the route reflector redistributes routes within the cluster, the BGP systems within the cluster do not have to be fully meshed.

When the route reflector receives a route, it selects the best path. Then, if the route came from a nonclient peer, the route reflector sends the route to all client peers within the cluster. If the route came from a client peer, the route reflector sends it to all nonclient peers and to all client peers except the originator. In this process, none of the client peers send routes to other client peers.

To configure route reflection, you specify a cluster identifier only on the BGP systems that are to be the route reflectors. These systems then determine, from the network reachability information they receive, which BGP systems are part of its cluster and are client peers, and which BGP systems are outside the cluster and are nonclient peers.

NOTE: When you configure route reflection on a Juniper Networks routing device, you can apply policy changes to the following attributes: NEXT_HOP, AS_PATH, LOCAL_PREF, and MED. Other vendors might not support policy changes to these attributes and so care must be taken with policy when migrating route reflection configurations from routing devices other than Juniper Networks devices to Juniper Networks devices.

To configure a routing device to be a route reflector, you must do the following:

- Configure multiple IBGP groups.
- Configure a cluster identifier (using the cluster statement) for groups that are members of the cluster.
- Configure all the groups with the same IBGP AS number.

To configure the route reflector, include the following statements in the configuration:

```
group group-name {
    type internal;
    peer-as autonomous-system;
    neighbor address1;
    neighbor address2;
}
group group-name {
    type internal;
    peer-as autonomous-system;
    cluster cluster-identifier;
```

```
    neighbor address3;
    neighbor address4;
}
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

By default, the BGP route reflector performs intracluster reflection because it assumes that all the client peers are not fully meshed. However, if the client peers are fully meshed, intracluster reflection results in the sending of redundant route advertisements. In this case, you can disable intracluster reflection by including the no-client-reflect statement within the group statement:

```
group group-name {
    type internal;
    peer-as autonomous-system;
    cluster cluster-identifier;
    no-client-reflect;
    neighbor address3;
    neighbor address4;
}
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

Examples: Configuring BGP Route Reflection

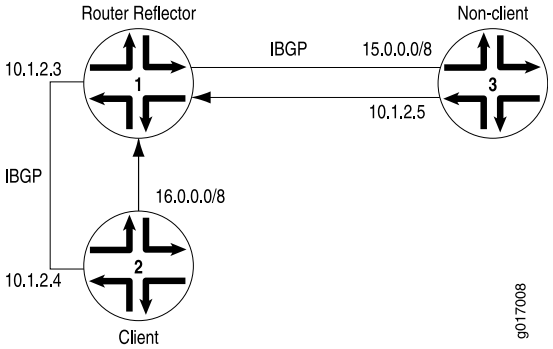
This example shows how to configure a simple route reflector. The configuration shown in Figure 12 on page 764 contains three routers: Router 1, which is the route reflector; Router 2, which is a client; and Router 3, which is a nonclient.

The routers have the following loopback addresses:

- Router 1—10.1.2.3
- Router 2—10.1.2.4
- Router 3—10.1.2.5

You must configure all routers to run a common IGP or to have static configuration, so that they learn each other's loopback addresses.

Figure 12: Simple Route Reflector



Configure Router 1 to be a route reflector for Router 2 and a regular IBGP neighbor for Router 3:

```
[edit]
routing-options {
   autonomous-system 65534;
}
protocols {
   bgp {
      group 13 {
         type internal;
         local-address 10.1.2.3;
         neighbor 10.1.2.5;
      }
      group 12 {
         type internal;
         local-address 10.1.2.3;
         cluster 1.2.3.4;
         neighbor 10.1.2.4;
      }
   }
}
```

Configure Router 2 to be an IBGP neighbor to Router 1 and announce 16.0.0.0/8 to Router 1. Configure route 16.0.0.0/8 as a static route on Router 2.

```
[edit]
routing-options {
   static {
      route 16.0.0.0/8 nexthop 172.16.1.2;
   }
   autonomous-system 65534;
}
protocols {
   bgp {
      group 21 {
         type internal;
         local-address 10.1.2.4;
         export dist-static;
         neighbor 10.1.2.3;
      }
```

```
    }
  }
  policy-options {
    policy-statement dist-static {
      from protocol static;
      then accept;
    }
  }
```

Configure Router 3 to be an IBGP neighbor to Router 1 and announce 15.0.0.0/8 to Router 1. Configure route 15.0.0.0/8 as a static route on Router 3.

```
[edit]
routing-options {
  static {
    route 15.0.0.0/8 nexthop 172.16.1.2;
  }
  autonomous-system 65534;
}
protocols {
  bgp {
    group 31 {
      type internal;
      local-address 10.1.2.5;
      export dist-static;
      neighbor 10.1.2.3;
    }
  }
}
policy-options {
  policy-statement dist-static {
    from protocol static;
    then accept;
  }
}
```

The following is the output of the show route detail command for route 16.0.0.0/8 on Router 1 and Router 3. Note that Router 1 learns 16.0.0.0/8 from its client, Router 2, and reflects it to Router 3. On Router 3, the output of the show route commands include the cluster list and originator ID attributes, which are added by Router 1 when the route is reflected.

Router 1

```
user@router1> show route 16.0.0.0/8 detail
inet.0: 1 destinations, 1 routes (1 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both
16.0.0.0/8 (1 entry, 1 announced)
*BGP   Preference: 170/-101
Source: 10.1.2.4
Nexthop: 172.16.1.2 via fxp0.0, selected
State: <Active Int Ext>
Local AS: 65534        Peer AS: 65534
Age: 11:55        Metric2: 0
Task: BGP_65534.10.1.2.4+4327
Announcement bits (3): 2-KRT 3-BGP.0.0.0.0+179 4-BGP_Sync_Any
AS path: I
BGP next hop: 172.16.1.2
```

```
Localpref: 100
Router ID: 10.1.2.4
```

Router 3

```
user@router3> show route 16.0.0.0/8 detail
inet.0: 1 destinations, 1 routes (1 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both
16.0.0.0/8 (1 entry, 1 announced)
*BGP   Preference: 170/-101
Source: 10.1.2.3
Nexthop: 172.16.1.2 via fxp0.0, selected
State: <Active Int Ext>
Local AS: 65534        Peer AS: 65534
Age: 11:57      Metric2: 0
Task: BGP_65534.10.1.2.3+4619
Announcement bits (2): 2-KRT 4-BGP_Sync_Any
AS path: I <Originator>
Cluster list: 1.2.3.4
Originator ID: 10.1.2.4
BGP next hop: 172.16.1.2
Localpref: 100
Router ID: 10.1.2.3
```

The following is the output of the show route detail command for route 15.0.0.0/8 on Router 1 and Router 2. Similar to when routes are reflected from client peers to nonclient peers, Rrouter 1 reflects a route it learns from a regular IBGP neighbor to its client. Cluster list and Originator ID attributes are added during the reflection process.

Router 1

```
user@router1> show route 15.0.0.0/8 detail
inet.0: 1 destinations, 1 routes (1 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both
15.0.0.0/8 (1 entry, 1 announced)
*BGP   Preference: 170/-101
Source: 10.1.2.5
Nexthop: 172.16.1.2 via fxp0.0, selected
State: <Active Int Ext>
Local AS: 65534        Peer AS: 65534
Age: 11:14        Metric2: 0
Task: BGP_65534.10.1.2.5+179
Announcement bits (3): 2-KRT 3-BGP.0.0.0.0+179 4-BGP_Sync_Any
AS path: I
BGP next hop: 172.16.1.2
Localpref: 100
Router ID: 10.1.2.5
```

Router 2

```
user@router2> show route 15.0.0.0/8 detail
inet.0: 1 destinations, 1 routes (1 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both
15.0.0.0/8 (1 entry, 1 announced)
*BGP   Preference: 170/-101
Source: 10.1.2.3
Nexthop: 172.16.1.2 via fxp0.0, selected
State: <Active Int Ext>
Local AS: 65534        Peer AS: 65534
Age: 11:23        Metric2: 0
Task: BGP_65534.10.1.2.3+179
Announcement bits (2): 2-KRT 4-BGP_Sync_Any
AS path: I <Originator>
Cluster list: 1.2.3.4
```

```
Originator ID: 10.1.2.5
BGP next hop: 172.16.1.2
Localpref: 100
Router ID: 10.1.2.3
```

Configuring Flap Damping for BGP Routes

BGP *route flapping* describes the situation in which BGP systems send an excessive number of update messages to advertise network reachability information. BGP *flap damping* is a method of reducing the number of update messages sent between BGP peers, thereby reducing the load on these peers, without adversely affecting the route convergence time for stable routes.

By default, route flap damping is disabled. To enable it, include the damping statement:

```
damping;
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

Damping is applied to external peers and to peers at confederation boundaries. For finer control over which peers have damping enabled, include the damping statement at the [edit protocols bgp group *group-name*] hierarchy level.

By default, route flap damping uses the following parameters:

- Decay half-life while reachable—15 minutes
- Maximum hold-down time—60 minutes
- Reuse threshold—750
- Cutoff threshold—3000

To change these default parameters, you must define the flap damping parameters by including the damping statement at the [edit policy-options] hierarchy level and then apply them by including an import statement when configuring BGP. For more information about flap damping and defining flap damping parameters, see the *JUNOS Policy Framework Configuration Guide*. For more information about applying policy filters in BGP, see "Applying Policies to BGP Routes" on page 781.

Enabling Multiprotocol BGP

Multiprotocol BGP (MP-BGP) is an extension to BGP that enables BGP to carry routing information for multiple network layers and address families. MP-BGP can carry the unicast routes used for multicast routing separately from the routes used for unicast IP forwarding.

To enable MP-BGP, you configure BGP to carry network layer reachability information (NLRI) for address families other than unicast IPv4 by including the family inet statement:

```
family inet {
    (any | flow | labeled-unicast | multicast | unicast) {
```

```
        accepted-prefix-limit {
            maximum number;
            teardown <percentage> <idle-timeout (forever | minutes)>;
        }
        <loops number>;
        prefix-limit {
            maximum number;
            teardown <percentage> <idle-timeout (forever | minutes)>;
        }
        rib-group group-name;
    }
}
```

To enable MP-BGP to carry NLRI for the IPv6 address family, include the **family inet6** statement:

```
family inet6 {
    (any | labeled-unicast | multicast | unicast) {
        accepted-prefix-limit {
            maximum number;
            teardown <percentage> <idle-timeout (forever | minutes)>;
        }
        <loops number>;
        prefix-limit {
            maximum number;
            teardown <percentage> <idle-timeout (forever | minutes)>;
        }
        rib-group group-name;
    }
}
```

On routers only, to enable MP-BGP to carry Layer 3 VPN NLRI for the IPv4 address family, include the **family inet-vpn** statement:

```
family inet-vpn {
    (any | flow | multicast | unicast) {
        accepted-prefix-limit {
            maximum number;
            teardown <percentage> <idle-timeout (forever | minutes)>;
        }
        <loops number>;
        prefix-limit {
            maximum number;
            teardown <percentage> <idle-timeout (forever | minutes)>;
        }
        rib-group group-name;
    }
}
```

On routers only, to enable MP-BGP to carry Layer 3 VPN NLRI for the IPv6 address family, include the **family inet6-vpn** statement:

```
family inet6-vpn {
    (any | multicast | unicast) {
        accepted-prefix-limit {
            maximum number;
```

```
        teardown <percentage> <idle-timeout (forever | minutes)>;
      }
      <loops number>;
      prefix-limit {
        maximum number;
        teardown <percentage> <idle-timeout (forever | minutes)>;
      }
      rib-group group-name;
    }
  }
```

On routers only, to enable MP-BGP to carry multicast VPN NLRI for the IPv4 address family and to enable VPN signaling, include the **family inet-mvpn** statement:

```
family inet-mvpn {
  signaling {
    accepted-prefix-limit {
      maximum number;
      teardown <percentage> <idle-timeout (forever | minutes)>;
    }
    <loops number>;
    prefix-limit {
      maximum number;
      teardown <percentage> <idle-timeout (forever | minutes)>;
    }
  }
}
```

To enable MP-BGP to carry multicast VPN NLRI for the IPv6 address family and to enable VPN signaling, include the **family inet6-mvpn** statement:

```
family inet6-mvpn {
  signaling {
    accepted-prefix-limit {
      maximum number;
      teardown <percentage> <idle-timeout (forever | minutes)>;
    }
    <loops number>;
    prefix-limit {
      maximum number;
      teardown <percentage> <idle-timeout <forever | minutes>;
    }
  }
}
```

For more information about multiprotocol BGP-based multicast VPNs, see the *JUNOS VPNs Configuration Guide* and the *JUNOS Multicast Protocols Configuration Guide*.

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for these statements.

NOTE: If you change the address family specified in the [edit protocols bgp family] hierarchy level, the BGP sessions are dropped and then reestablished.

In JUNOS Release 9.6 and later, you can specify a loops value for a specific BGP address family. Include the loops *number* statement to specify the maximum number of times that the AS number can appear in the AS path advertised by a BGP peer for a specific address family. When you specify the loops statement for a BGP address family, that value is used to determine maximum number of times the AS number can appear in the AS path rather than any loops value configured for the global AS. For *number*, specify a value from 1 through 10.

By default, BGP peers carry only unicast routes used for unicast forwarding purposes. To configure BGP peers to carry only multicast routes, specify the multicast option. To configure BGP peers to carry both unicast and multicast routes, specify the any option.

When MP-BGP is configured, BGP installs the MP-BGP routes into different routing tables. Each routing table is identified by the protocol family or address family indicator (AFI) and a subaddress family indicator (SAFI).

The JUNOS Software supports all unicast and multicast SAFIs (1 and 2) for both AFI 1 (IPv4) and AFI 2 (IPv6). The following table shows all possible AFI and SAFI combinations and routing tables populated with this information:

	SAFI 1	SAFI 2
AFI 1 (IPv4)	inet.0	inet.2
AFI 2 (IPv6)	inet6.0	inet6.2

Routes installed in the inet.2 routing table can only be exported to MP-BGP peers because they use sub-address family information (SAFI) identifying them as routes to multicast sources. Routes installed in the inet.0 routing table can only be exported to standard BGP peers.

The inet.2 routing table should be a subset of the routes that you have in inet.0, since it is unlikely that you would have a route to a multicast source to which you could not send unicast traffic. The inet.2 routing table stores the unicast routes that are used for multicast reverse-path-forwarding checks and the additional reachability information learned by MP-BGP from the NLRI multicast updates. An inet.2 routing table is automatically created when you configure MP-BGP (by setting NLRI to any).

When you enable multiprotocol BGP, you can do the following:

- Limiting the Number of Prefixes Received on a BGP Peering Session on page 771
- Limiting the Number of Prefixes Accepted on a BGP Peering Session on page 771
- Configuring BGP Routing Table Groups on page 772
- Resolving Routes to PE Routing Devices Located in Other ASs on page 772
- Allowing Labeled and Unlabeled Routes on page 773

Limiting the Number of Prefixes Received on a BGP Peering Session

You can limit the number of prefixes received on a BGP peering session, and log rate-limited messages when the number of injected prefixes exceeds a set limit. You can also tear down the peering when the number of prefixes exceeds the limit.

To configure a limit to the number of prefixes that can received on a BGP session, include the prefix-limit statement:

```
prefix-limit {
    maximum number;
    teardown <percentage> <idle-timeout (forever | minutes)>;
}
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

For maximum *number*, specify a value in the range from 1 through 4,294,967,295. When the specified maximum number of prefixes is exceeded, a system log message is sent.

If you include the teardown statement, the session is torn down when the maximum number of prefixes is exceeded. If you specify a percentage, messages are logged when the number of prefixes exceeds that percentage of the specified maximum limit. After the session is torn down, it is reestablished in a short time (unless you include the idle-timeout statement). Then the session can be kept down for a specified amount of time, or forever. If you specify forever, the session is reestablished only after the you issue a clear bgp neighbor command.

NOTE: In JUNOS Release 9.2 and later, you can alternatively configure a limit to the number of prefixes that can accepted on a BGP peering session. For more information, see “Limiting the Number of Prefixes Accepted on a BGP Peering Session” on page 771.

Limiting the Number of Prefixes Accepted on a BGP Peering Session

In JUNOS Release 9.2 and later, you can limit the number of prefixes that can be accepted on a BGP peering session. When that specified limit is exceeded, a system log message is sent. You can also specify to reset the BGP session if the limit to the number of specified prefixes is exceeded.

To configure a limit to the number of prefixes that can be accepted on a BGP peering session, include the accepted-prefix-limit statement:

```
accepted-prefix-limit {
    maximum number;
    teardown <percentage> <idle-timeout (forever | minutes)>;
}
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

For maximum *number*, specify a value in the range from 1 through 4,294,967,295.

Include the teardown statement to specify to the reset the BGP peering session when the number of accepted prefixes exceeds the configured limit. You can also include a percentage value from 1 through 100 to have a system log message sent when the number of accepted prefixes exceeds that percentage of the maximum limit. By default a BGP session that is reset is reestablished within a short time. Include the idle-timeout statement to prevent the BGP session from being reestablished for a specified period of time. You can configure a timeout value from 1 through 2400 minutes. Include the forever option to prevent the BGP session from being reestablished until you issue the clear bgp neighbor command.

NOTE: When nonstop active routing (NSR) is enabled and a switchover to a backup Routing Engine occurs, BGP peers that are down are automatically restarted. The peers are restarted even if the idle-timeout forever statement is configured.

NOTE: Alternatively, you can instead configure a limit to the number of prefixes that can be *received* (as opposed to accepted) on a BGP peering session. For more information, see "Limiting the Number of Prefixes Received on a BGP Peering Session" on page 771.

Configuring BGP Routing Table Groups

When a BGP session receives a unicast or multicast NLRI, it installs the route in the appropriate table (inet.0 or inet6.0 for unicast, and inet.2 or inet6.2 for multicast). To add unicast prefixes to both the unicast and multicast tables, you can configure BGP routing table groups. This is useful if you cannot perform multicast NLRI negotiation.

To configure BGP routing table groups, include the rib-group statement:

```
rib-group group-name;
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

Resolving Routes to PE Routing Devices Located in Other ASs

You can allow labeled routes to be placed in the inet.3 routing table for route resolution. These routes are then resolved for provider edge (PE) routing device connections where the remote PE is located across another AS. For a PE routing device to install a route in the VRF routing instance, the next hop must resolve to a route stored within the inet.3 table.

To resolve routes into the inet.3 routing table, include the resolve-vpn statement:

resolve-vpn *group-name*;

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

Allowing Labeled and Unlabeled Routes

You can allow both labeled and unlabeled routes to be exchanged in a single session. The labeled routes are placed in the inet.3 routing table, and both labeled and unlabeled unicast routes can be sent or received by the routing device.

To allow both labeled and unlabeled routes to be exchanged, include the rib inet.3 statement:

rib inet.3;

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

Enabling BGP to Carry Flow-Specification Routes

You can allow BGP to carry flow-specification NLRI messages. Flow routes are encapsulated into the flow-specification NLRI and propagated through a network or VPNs, sharing filter-like information. Flow routes are an aggregation of match conditions and resulting actions for packets. They provide you with traffic filtering and rate-limiting capabilities much like firewall filters.

When you enable flow-specification routes, you can do the following:

- Configuring Flow-Specification Routes for IPv4 Unicast on page 773
- Configuring Flow-Specification Routes for Layer 3 VPNs on page 774

Configuring Flow-Specification Routes for IPv4 Unicast

To enable MP-BGP to carry flow-specification NLRI for the inet address family, include the flow statement:

flow;

NOTE: Unicast flow routes are supported for the default instance, VRF instances, and virtual-router instances only. Instance type is configured by including the instance-type statement at the [edit routing-instance *instance-name*] hierarchy level.

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

Flow routes received using the BGP NLRI messages are validated before they are installed into the flow routing table *instance-name*.inetflow.0. The validation procedure is described in the Internet draft draft-ietf-idr-flow-spec-09.txt, *Dissemination of Flow*

Specification Rules. You can bypass the validation process and use your own specific import policy.

To disable the validation procedure and use an import policy instead, include the no-validate statement at the [edit protocols bgp group *group-name* family inet flow] hierarchy level:

```
no-validate policy-name;
```

Configuring Flow-Specification Routes for Layer 3 VPNs

On routers only, the VPN compares the route target extended community in the NLRI to the import policy. If there is a match, the VPN can start using the flow routes to filter and rate-limit packet traffic. Received flow routes are installed into the flow routing table *instance-name*.inetflow.0.

Flow routes can also be propagated throughout a VPN network and shared among VPNs, providing filter and rate-limiting capabilities.

To enable MP-BGP to carry flow-specification NLRI for the inet-vpn address family, include the flow statement at the [edit protocols bgp group *group-name* family inet-vpn] hierarchy level:

```
flow;
```

NOTE: VPN flow routes are supported for the default instance only. Instance type is configured by including the instance-type statement at the [edit routing-instance *instance-name*] hierarchy level.

Flow routes configured for VPNs with family inet-vpn are not automatically validated, so the no-validate statement is not supported at the [edit protocols bgp group *group-name* family inet-vpn] hierarchy level.

For more information on flow routes, see "Configuring Flow Routes" on page 105 and the Internet draft draft-marques-idr-flow-spec-09.txt, *Dissemination of Flow Specification Rules*.

Enabling BGP to Carry CLNS Routes

Connectionless Network Service (CLNS) is a Layer 3 protocol similar to IP version 4 (IPv4). CLNS uses network service access points (NSAPs) to address end systems. This allows for a seamless AS based on ISO NSAPs.

NOTE: CLNS is supported for the J Series Services Router only.

A single routing domain consisting of ISO NSAP devices are considered to be CLNS islands. CLNS islands are connected together by VPNs.

You can configure BGP to exchange ISO CLNS routes between PE routers connecting various CLNS islands in a VPN using multiprotocol BGP extensions. These extensions are the ISO VPN NLRIs.

To enable MP-BGP to carry CLNS VPN NLRIs, include the iso-vpn statement:

```
iso-vpn {
    unicast {
        prefix-limit number;
        rib-group group-name;
    }
}
```

To limit the number of prefixes from a peer, include the prefix-limit statement. To specify a routing table group, include the rib-group statement.

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

Each CLNS network island is treated as a separate VRF instance on the PE router.

You can configure CLNS on the global level, group level, and neighbor level.

For information on CLNS, see "Configuring CLNS for IS-IS" on page 356 and the *JUNOS Software Interfaces and Routing Configuration Guide*.

For sample configurations, see the following sections:

- Example: Enabling CLNS Between Two Routers on page 775
- Example: Configuring CLNS Within a VPN on page 777

Example: Enabling CLNS Between Two Routers

Configure CLNS between two routers through a route reflector:

On Router 1:

```
[edit protocols bgp]
protocols {
    bgp {
        local-address 10.255.245.195;
        group pe-pe {
            type internal;
            neighbor 10.255.245.194 {
                family iso-vpn {
                    unicast;
                }
            }
        }
    }
}
[edit routing-instances]
routing-instances {
    aaaa {
        instance-type vrf;
```

```
        interface fe-0/0/0.0;
        interface so-1/1/0.0;
        interface lo0.1;
        route-distinguisher 10.255.245.194:1;
        vrf-target target:11111:1;
        protocols {
            isis {
                export dist-bgp;
                no-ipv4-routing;
                no-ipv6-routing;
                clns-routing;
                interface all;
            }
        }
    }
}
```

On Router 2:

```
[edit protocols bgp]
protocols {
    bgp {
        group pe-pe {
            type internal;
            local-address 10.255.245.198;
            family route-target;
            neighbor 10.255.245.194 {
                family iso-vpn {
                    unicast;
                }
            }
        }
    }
}
[edit routing-instances]
routing-instances {
    aaaa {
        instance-type vrf;
        interface lo0.1;
        interface so-0/1/2.0;
        interface so-0/1/3.0;
        route-distinguisher 10.255.245.194:1;
        vrf-target target:11111:1;
        routing-options {
            rib aaaa.iso.0 {
                static {
                    iso-route 47.0005.80ff.f800.0000.bbbb.1022/104 next-hop
                        47.0005.80ff.f800.0000.aaaa.1000.1921.6800.4196.00;
                }
            }
        }
        protocols {
            isis {
                export dist-bgp;
                no-ipv4-routing;
                no-ipv6-routing;
                clns-routing;
                interface all;
```

```
        }
      }
    }
}
On Route Reflector:
[edit protocols bgp]
protocols {
    bgp {
        group pe-pe {
            type internal;
            local-address 10.255.245.194;
            family route-target;
            neighbor 10.255.245.195 {
                cluster 0.0.0.1;
            }
            neighbor 10.255.245.198 {
                cluster 0.0.0.1;
            }
        }
    }
}
```

Example: Configuring CLNS Within a VPN

Configure CLNS on three PE routers within a VPN:

```
On PE Router 1:
[edit protocols bgp]
protocols {
    mpls {
        interface all;
    }
    bgp {
        group asbr {
            type external;
            local-address 10.245.245.3;
            neighbor 10.245.245.1 {
                multihop;
                family iso-vpn {
                    unicast;
                }
                peer-as 200;
            }
        }
    }
}
[edit routing-instances]
routing-instances {
    aaaa {
        instance-type vrf;
        interface lo0.1;
        interface t1-3/0/0.0;
        interface fe-5/0/1.0;
        route-distinguisher 10.245.245.1:1;
        vrf-target target:11111:1;
```

```
        protocols {
            isis {
                export dist-bgp;
                no-ipv4-routing;
                no-ipv6-routing;
                clns-routing;
                interface all;
            }
        }
    }
}
```

On PE Router 2:

```
[edit protocols bgp]
protocols {
    bgp {
        group asbr {
            type external;
            multihop;
            family iso-vpn {
                unicast;
            }
            neighbor 10.245.245.2 {
                peer-as 300;
            }
            neighbor 10.245.245.3 {
                peer-as 100;
            }
        }
    }
}
[edit routing-instances]
routing-instances {
    aaaa {
        instance-type vrf;
        interface lo0.1;
        route-distinguisher 10.245.245.1:1;
        vrf-target target:11111:1;
    }
}
```

On PE Router 3:

```
[edit protocols bgp]
protocols {
    bgp {
        group asbr {
            type external;
            multihop;
            local-address 10.245.245.2;
            neighbor 10.245.245.1 {
                family iso-vpn {
                    unicast;
                }
                peer-as 200;
            }
        }
    }
}
```

```
[edit routing-instances]
routing-instances {
   aaaa {
      instance-type vrf;
      interface lo0.1;
      interface fe-0/0/1.0;
      interface t1-3/0/0.0;
      route-distinguisher 10.245.245.1:1;
      vrf-target target:11111:1;
      protocols {
         isis {
            export dist-bgp;
            no-ipv6-routing;
            clns-routing;
            interface all;
         }
      }
   }
}
```

Enabling BGP Route Target Filtering

You can limit the number of prefixes advertised on BGP peerings specifically to the peers that need the updates.

In a VPN provider network, a BGP speaker advertises all VPN routes to the peers in the same VPN. Peers that are configured either as a route reflector or border router for a VPN must store all routes within the network. While PE routers automatically discard routes that do not affect them, these route updates must still be generated and received.

Enabling route target filtering allows you to limit these route updates.

To enable route target filtering, include the route-target statement:

```
route-target {
   advertise-default;
   external-paths number;
   prefix-limit {
      maximum number;
      teardown <percentage> <idle-timeout (forever | minutes)>;
   }
}
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

If you include the advertise-default statement, the router advertises the default route-target route (0:0:0/0) and suppresses any specific route-target routes. This is useful for a route reflector in a BGP group consisting of neighbor PE routers only. If you include the external-paths statement, the router limits the number of external paths accepted for route filtering. The range is from 1 through 16. The default is 1. If you include the teardown statement, the session is torn down when the maximum number of prefixes is reached. If you specify a percentage, messages are logged

when the number of prefixes reaches that percentage. Once the session is torn down, it is reestablished in a short time. Include the idle-timeout statement to keep the session down for a specified amount of time, or forever. If you specify forever, the session is reestablished only after you use the clear bgp neighbor command.

For more information about VPNs, see the *JUNOS VPNs Configuration Guide*.

Applying Filters Provided by BGP Peers to Outbound Routes

You can configure a BGP peer to accept route filters from remote peers and perform outbound route filtering using the received filters. By filtering out unwanted updates, the sending peer saves resources needed to generate and transmit updates, and the receiving peer saves resources needed to process updates. This feature can be useful, for example, in a virtual private network (VPN) in which subsets of customer edge (CE) devices are not capable of processing all the routes in the VPN. The CE's can use prefix-based outbound route filtering to communicate to the provider edge (PE) routing device to transmit only a subset of routes, such as routes to the main data centers only.

To configure prefix-based outbound route filtering, include the following statements:

```
outbound-route-filter {
    bgp-orf-cisco-mode;
    prefix-based {
        accept {
            (inet | inet6);
        }
    }
}
```

For a complete list of hierarchy levels at which you can configure these statements, see the statement summaries for these statements.

NOTE: The maximum number of prefix-based outbound route filters that a BGP peer can accept is 5000. If a remote peer sends more than 5000 outbound route filters to a peer address, the additional filters are discarded and a system log message is generated.

You can also enable interoperability with routing devices that use the vendor-specific compatibility code of 130 for outbound route filters and the code type of 128. The standard code is 3, and the standard code type is 64. You can configure interoperability for the routing device as a whole or for specific BGP groups or peers only.

To configure BGP peers to interoperate with routing devices that use vendor-specific compatibility codes for outbound route filters, include the bgp-orf-cisco-mode statement:

```
outbound-route-filter {
    bgp-orf-cisco-mode;
}
```

Enabling Layer 2 VPN and VPLS Signaling

You can enable BGP to carry Layer 2 VPN and VPLS NLRI messages.

To enable VPN and VPLS signaling, include the **family** statement:

```
family {
    l2vpn {
        signaling {
            prefix-limit {
                maximum number;
                teardown <percentage> <idle-timeout (forever | minutes)>;
            }
        }
    }
}
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

To configure a maximum number of prefixes, include the **prefix-limit** statement:

```
prefix-limit {
    maximum number;
    teardown <percentage> <idle-timeout (forever | minutes)>;
}
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

When you set the maximum number of prefixes, a message is logged when that number is reached. If you include the **teardown** statement, the session is torn down when the maximum number of prefixes is reached. If you specify a percentage, messages are logged when the number of prefixes reaches that percentage. Once the session is torn down, it is reestablished in a short time. Include the **idle-timeout** statement to keep the session down for a specified amount of time, or forever. If you specify **forever**, the session is reestablished only after you use the **clear bgp neighbor** command.

For more information about VPNs, see the *JUNOS VPNs Configuration Guide*. For a detailed VPLS example configuration, see the *JUNOS Feature Guide*.

Applying Policies to BGP Routes

All routing protocols use the JUNOS Software routing table to store the routes that they learn and to determine which routes they should advertise in their protocol packets. Routing policy allows you to control which routes the routing protocols store in and retrieve from the routing table. For information about routing policy, see the *JUNOS Policy Framework Configuration Guide*.

When configuring BGP routing policy, you can perform the following tasks:

- Applying Routing Policy on page 782
- Setting BGP to Advertise Inactive Routes on page 783
- Configuring BGP to Advertise the Best External Route to Internal Peers on page 783
- Configuring How Often BGP Exchanges Routes with the Routing Table on page 784
- Disabling Suppression of Route Advertisements on page 785

Applying Routing Policy

You define routing policy at the [edit policy-options] hierarchy level. To apply policies you have defined for BGP, include the import and export statements within the BGP configuration. For information about defining policy, see the *JUNOS Policy Framework Configuration Guide*.

You can apply policies as follows:

- BGP global import and export statements—Include these statements at the [edit protocols bgp] hierarchy level (for routing instances, include these statements at the [edit routing-instances *routing-instance-name* protocols bgp] hierarchy level).
- Group import and export statements—Include these statements at the [edit protocols bgp group *group-name*] hierarchy level (for routing instances, include these statements at the [edit routing-instances *routing-instance-name* protocols bgp group *group-name*] hierarchy level).
- Peer import and export statements—Include these statements at the [edit protocols bgp group *group-name* neighbor *address*] hierarchy level (for routing instances, include these statements at the [edit routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address*] hierarchy level).

A peer-level import or export statement overrides a group import or export statement. A group-level import or export statement overrides a global BGP import or export statement.

To apply policies, see the following sections:

- Applying Policies to Routes Being Imported into the Routing Table from BGP on page 782
- Applying Policies to Routes Being Exported from the Routing Table into BGP on page 783

Applying Policies to Routes Being Imported into the Routing Table from BGP

To apply policy to routes being imported into the routing table from BGP, include the import statement, listing the names of one or more policies to be evaluated:

```
import [ policy-names ];
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

If you specify more than one policy, they are evaluated in the order specified, from first to last, and the first matching filter is applied to the route. If no match is found, BGP places into the routing table only those routes that were learned from BGP routing devices.

Applying Policies to Routes Being Exported from the Routing Table into BGP

To apply policy to routes being exported from the routing table into BGP, include the export statement, listing the names of one or more policies to be evaluated:

```
export [ policy-names ];
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

If you specify more than one policy, they are evaluated in the order specified, from first to last, and the first matching filter is applied to the route. If no routes match the filters, the routing table exports into BGP only the routes that it learned from BGP.

Setting BGP to Advertise Inactive Routes

By default, BGP stores the route information it receives from update messages in the JUNOS routing table, and the routing table exports only active routes into BGP, which BGP then advertises to its peers. To have the routing table export to BGP the best route learned by BGP even if the JUNOS Software did not select it to be an active route, include the advertise-inactive statement:

```
advertise-inactive;
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

Configuring BGP to Advertise the Best External Route to Internal Peers

In general, deployed BGP implementations do not advertise the external route with the highest local preference value to internal peers unless it is the best route. Although this behavior was required by an earlier version of the BGP version 4 specification, RFC 1771, it was typically not followed in order to minimize the amount of advertised information and to prevent routing loops. However, there are scenarios in which advertising the best external route is beneficial, in particular, situations that can result in IBGP route oscillation.

In JUNOS Release 9.3 and later, you can configure BGP to advertise the best external route into an IBGP mesh group, a route reflector cluster, or an AS confederation, even when the best route is an internal route.

NOTE: In order to configure the advertise-external statement on a route reflector, you must disable intracluster reflection with the no-client-reflect statement.

When a routing device is configured as a route reflector for a cluster, a route advertised by the route reflector is considered internal only if it is received from an internal peer with the same cluster identifier or no cluster identifier. A route received from an internal peer that belongs to a another cluster, that is, with a different cluster identifier, is considered external.

In a confederation, when advertising a route to a confederation border router, any route from a different confederation sub-AS is considered internal.

You can also configure BGP only to advertise the external route if the route selection process reaches the point where the multiple exit discriminator (MED) metric is evaluated. As a result, an external route with an AS path worse (that is, longer) than that of the active path is not advertised.

The JUNOS Software also provides support for configuring a BGP export policy that matches on the state of an advertised route. You can match on either active or inactive routes. For more information, see the *JUNOS Policy Framework Configuration Guide*.

To configure BGP to advertise the best external path to internal peers, include the advertise-external statement:

```
advertise-external;
```

NOTE: The advertise-external statement is supported at both the group and neighbor level. If you configure the statement at the neighbor level, you must configure it for all neighbors in a group. Otherwise, the group is automatically split into different groups.

For a complete list of hierarchy levels at which you can configure this statement, see the statement summary section for this statement.

To configure BGP to advertise the best external path only if the route selection process reaches the point where the MED is evaluated, include the conditional statement:

```
advertise-external {
    conditional;
}
```

For a complete list of hierarchy levels at which you can configure this statement, see the statement summary section for this statement.

Configuring How Often BGP Exchanges Routes with the Routing Table

BGP stores the route information it receives from update messages in the routing table, and the routing table exports active routes from the routing table into BGP. BGP then advertises the exported routes to its peers. By default, the exchange of route information between BGP and the routing table occurs immediately after the routes are received. This immediate exchange of route information might cause instabilities in the network reachability information. To guard against this, you can delay the time between when BGP and the routing table exchange route information.

To configure how often BGP and the routing table exchange route information, include the out-delay statement:

```
out-delay seconds;
```

By default, the routing table retains some of the route information learned from BGP. To have the routing table retain all or none of this information, include the keep statement:

```
keep (all | none);
```

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for these statements.

The routing table can retain the route information learned from BGP in one of the following ways:

- Default (omit the keep statement)—Keep all route information that was learned from BGP except for routes whose AS path is looped and the loop includes the local AS.
- keep all—Keep all route information that was learned from BGP.
- keep none—Discard routes that were received from a peer and that were rejected by import policy or other sanity checking, such as AS path or next hop. When you configure keep none for the BGP session and the inbound policy changes, the JUNOS Software forces readvertisement of the full set of routes advertised by the peer.

In an AS path healing situation, routes with looped paths theoretically could become usable during a soft reconfiguration when the AS path loop limit is changed. However, there is a significant memory usage difference between the default and keep all because it is common for a peer to readvertise routes back to the peer from which it learned them. The default behavior is not to waste memory on such routes.

Disabling Suppression of Route Advertisements

The JUNOS Software does not advertise the routes learned from one EBGP peer back to the same EBGP peer. In addition, the software does not advertise those routes back to any EBGP peers that are in the same AS as the originating peer, regardless of the routing instance. You can modify this behavior by including the advertise-peer-as statement in the configuration. To disable the default advertisement suppression, including the advertise-peer-as statement:

```
advertise-peer-as;
```

NOTE: The route suppression default behavior is disabled if the as-override statement is included in the configuration.

If you include the advertise-peer-as statement in the configuration, BGP advertises the route regardless of this check.

To restore the default behavior, include the no-advertise-peer-as statement in the configuration:

```
no-advertise-peer-as;
```

If you include both the as-override and no-advertise-peer-as statements in the configuration, the no-advertise-peer-as statement is ignored. You can include these statements at multiple hierarchy levels.

For a list of hierarchy levels at which you can include these statements, see the statement summary section for this statement.

Preventing Automatic Reestablishment of BGP Peering Sessions After NSR Switchovers

It is useful to prevent a BGP peering session from automatically being reestablished after a nonstop active routing (NSR) switchover when you have applied routing policies configured in the dynamic database. When NSR is enabled, the dynamic database is not synchronized with the backup Routing Engine. Therefore, when a switchover occurs, import and export policies configured in the dynamic database might no longer be available. For more information about configuring dynamic routing policies, see the *JUNOS Policy Framework Configuration Guide*.

You can configure the routing device not to reestablish a BGP peering session after an NSR switchover either for a specified period or until you manually reestablish the session. Include the idle-after-switch-over statement at the [edit protocols bgp] hierarchy level:

```
idle-after-switch-over (forever | seconds);
```

For a list of hierarchy levels at which you can configure this statement, see the configuration statement summary for this statement.

For *seconds*, specify a value from 1 through 4294967295. The BGP peering session is not reestablished until after the specified period. If you specify the forever option, the BGP peering session is not reestablished until you issue the clear bgp neighbor command.

Configuring EBGP Peering Using IPv6 Link-Local Addresses

The JUNOS Software supports EBGP peering sessions by means of IPv6 link-local addresses. An IPv6 peering session can be configured when a 128-bit IPv6 address is specified in the neighbor statement. The peer address is identified as link-local by means of the local-interface statement.

To configure an EBGP peer, specify a 128-bit IPv6 link-local address in the neighbor statement:

```
neighbor ipv6-link-local-address;
```

To specify the interface name for the EBGP link-local peer, include the local-interface statement:

local-interface *interface-name*;

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for this statements.

This statement is valid only for 128-bit IPv6 link-local addresses and is mandatory for configuring an IPv6 EBGP link-local peering session. For more information about IPv6 addressing, see "Routing Protocols Concepts" on page 3.

NOTE: Configuring EBGP peering using link-local addresses is only applicable for directly connected interfaces. There is no support for multihop peering.

Configuring IPv6 BGP Routes over IPv4 Transport

You can export both IPv6 and IPv4 prefixes over an IPv4 connection where both sides are configured with an IPv4 interface. In this case, the BGP neighbors are IPv4 prefixes. The IPv4-compatible IPv6 prefixes are configured on the interfaces to preclude the configuration of static routes.

Keep the following in mind when exporting IPv6 BGP prefixes:

- BGP derives next-hop prefixes using the IPv4-compatible IPv6 prefix. For example, the IPv4 next-hop prefix 10.19.1.1 translates to the IPv6 next-hop prefix ::10.19.1.1 (hexadecimal format ::a13:101).

NOTE: There must be an active route to the IPv4-compatible IPv6 next hop to export IPv6 BGP prefixes.

- An IPv6 connection must be configured over the link. The connection must be either an IPv6 tunnel or a dual-stack configuration.
- When configuring IPv4-compatible IPv6 prefixes, use a mask that is longer than 96 bits.
- Configure a static route if you want to use normal IPv6 prefixes.

Example: Configuring IPv6 BGP Routes over IPv4 Transport

Configure IPv4 transport from interface ge-0/1/0 with an IPv4 prefix 11.19.1.2/24 to interface ge-1/1/1 with an IPv4 prefix 11.19.1.1/24 to carry IPv6 BGP routes.

Define IPv4 and IPv6 BGP groups for 11.19.1.2 with BGP neighbor 11.19.1.1:

```
[edit protocols]
bgp {
   group ebgp_both {
      type external;
      local-address 11.19.1.2;
      family inet {
         unicast;
```

```
        }
        family inet6 {
            unicast;
        }
        peer-as 1;
        neighbor 11.19.1.1;
    }
}
```

Configure the interfaces with both an IPv4 and a corresponding IPv4-compatible IPv6 prefix:

```
[edit interfaces]
ge-0/1/0 {
    unit 0 {
        family inet {
            address 11.19.1.2/24;
        }
        family inet6 {
            address ::11.19.1.2/126;
        }
    }
}
```

Configuring System Logging of BGP Peer State Transitions

Whenever a BGP peer makes a state transition, you can configure BGP so that it generates a syslog message. To do this, include the log-updown statement:

```
log-updown;
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

NOTE: Enabling the log-updown statement causes BGP state transitions to be logged at warning level.

Configuring a Text Description for BGP Groups or Peers

You can enter plain text to describe the BGP router configuration.

To enter a description, include the description statement:

```
description description-text;
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

Restricting TCP Connections to BGP Peers

You can restrict TCP connection attempts on port 179 to BGP peers only. This blocks all non-BGP connection attempts on port 179.

To restrict TCP connection attempts to BGP peers include the apply-path statement at the [edit policy-options prefix-list *list-name*] hierarchy level:

```
[edit policy-options prefix-list list-name]
apply-path protocol bgp group group-name neighbor neighbor;
```

For detailed information about configuring TCP connection attempts, see the *JUNOS Policy Framework Configuration Guide*.

Applying BGP Export Policy to VRF Routes

You can apply a VPN routing and forwarding (VRF) export policy in addition to applying a BGP export policy to routes before advertising the routes to provider edge (PE) routers in a VPN. The default action is to accept routes.

To apply an export policy to routes, include the vpn-apply-export statement:

```
vpn-apply-export;
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

Including Next-Hop Reachability Information in Multiprotocol Updates

To enable multiprotocol updates to contain next-hop reachability information, include the include-mp-next-hop statement:

```
include-mp-next-hop;
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

Configuring BFD for BGP

The Birectional Forwarding Detection (BFD) protocol is a simple hello mechanism that detects failures in a network. Hello packets are sent at a specified, regular interval. A neighbor failure is detected when the routing device stops receiving a reply after a specified interval. BFD works with a wide variety of network environments and topologies. The failure detection timers for BFD have shorter time limits than default failure detection mechanisms, providing faster detection. These timers are also adaptive and can be adjusted to be more or less aggressive. For example, the timers can adapt to a higher value if the adjacency fails, or a neighbor can negotiate a higher value for a timer than the one configured.

NOTE: In JUNOS Release 8.3 and later, BFD is supported on IBGP and multihop EBGP sessions as well as on single-hop EBGP sessions. BFD does not support IPv6 interfaces with BGP. In JUNOS Release 9.1 and later, BFD supports IPv6 interfaces in static routes only.

To enable failure detection, include the bfd-liveness-detection statement:

```
bfd-liveness-detection {
    detection-time {
        threshold milliseconds;
    }
    holddown-interval milliseconds;
    minimum-interval milliseconds;
    minimum-receive-interval milliseconds;
    no-adaptation;
    transmit-interval {
        threshold milliseconds;
        minimum-interval milliseconds;
    }
    multiplier number;
    no-adaptation;
    version (1 | automatic);
}
```

To specify the threshold for the adaptation of the detection time, include the detection-time threshold statement:

```
detection-time {
    threshold milliseconds;
}
```

When the BFD session detection time adapts to a value equal to or higher than the threshold, a single trap and a system log message are sent.

In JUNOS Release 8.5 and later, you can configure an interval to specify how long the BFD session for an EBGP peer must remain up before a state change notification is sent. When you configure the hold-down interval for the BFD protocol for EBGP, the BGP session goes down if the BFD session goes down. If you do not configure the BFD hold-down interval, the BGP session remains up even if the BFD session goes down. To specify the hold-down interval, include the holddown-interval statement:

```
holddown-interval milliseconds;
```

You can configure a value in the range from 0 through 255,000 and the default is 0. The holddown-interval statement is supported only for EBGP peers at the [edit protocols bgp group *group-name* neighbor *address*] hierarchy level. If the BFD session goes down and then comes back up during the configured hold-down interval, the timer is restarted.

NOTE: You must configure the hold-down interval on both EBGP peers.

NOTE: If you configure the hold-down interval for a multihop EBGP session, you must also configure a local IP address by including the local-address statement at the [edit protocols bgp group *group-name*] hierarchy level. For more information about configuring an EBGP multihop session, see "Configuring EBGP Multihop Sessions" on page 750. For more information about configuring the local address, see "Configuring a Local Endpoint Address for BGP Sessions" on page 746.

To specify the minimum transmit and receive intervals for failure detection, include the minimum-interval statement:

```
minimum-interval milliseconds;
```

This value represents the minimum interval at which the local routing device transmits hello packets as well as the minimum interval that the routing device expects to receive a reply from a neighbor with which it has established a BFD session. You can configure a value in the range from 1 through 255,000 milliseconds. You can also specify the minimum transmit and receive intervals separately.

NOTE: If you specify a minimum interval for Bidirectional Forwarding Detection (BFD) less than 100 ms, the BFD session might transition down and up. On some routing devices, it is safe to configure values below 100 ms. Please contact Juniper Networks customer support for more information.

To specify only the minimum receive interval for failure detection, include the minimum-receive-interval statement:

```
minimum-receive-interval milliseconds;
```

This value represents the minimum interval at which the local routing device expects to receive a reply from a neighbor with which it has established a BFD session. The values that you can configure are in the range from 1 through 255,000 milliseconds.

To specify the number of hello packets not received by a neighbor that causes the originating interface to be declared down, include the multiplier statement:

```
multiplier number;
```

The default is 3, and you can configure a value in the range from 1 through 255.

To specify only the minimum transmit interval for failure detection, include the transmit-interval minimum-interval statement:

```
transmit-interval {
    minimum-interval milliseconds;
}
```

This value represents the minimum interval at which the local routing device transmits hello packets to the neighbor with which it has established a BFD session. You can configure a value in the range from 1 through 255,000 milliseconds.

To specify the threshold for detecting the adaptation of the transmit interval, include the transmit-interval threshold statement:

```
transmit-interval {
   threshold milliseconds;
}
```

The threshold value must be greater than the transmit interval.

To specify the BFD version used for detection, include the version statement:

```
version (1 | automatic);
```

The default is to have the version detected automatically.

You can trace BFD operations by including the traceoptions statement at the [edit protocols bfd] hierarchy level. For more information, see "Tracing BFD Protocol Traffic" on page 78.

In JUNOS Release 9.0 and later, you can configure BFD sessions not to adapt to changing network conditions. To disable BFD adaptation, include the no-adaptation statement:

```
no-adaptation;
```

NOTE: We recommend that you not disable BFD adaptation unless it is preferable not to have BFD adaptation enabled in your network.

For a list of hierarchy levels at which you can include these statements, see the statement summary sections for these statements.

Overview of BFD Authentication for BGP

Bidirectional Forwarding Detection protocol (BFD) enables rapid detection of communication failures between adjacent systems. By default, authentication for BFD sessions is disabled. However, when you run BFD over Network Layer protocols, the risk of service attacks can be significant. We strongly recommend using authentication if you are running BFD over multiple hops or through insecure tunnels. Beginning with JUNOS Release 9.6, the JUNOS Software supports authentication for BFD sessions running over BGP. BFD authentication is not supported on MPLS OAM sessions. BFD authentication is only supported in the domestic image and is not available in the export image.

You authenticate BFD sessions by specifying an authentication algorithm and keychain, and then associating that configuration information with a security authentication keychain using the keychain name.

The following sections describe the supported authentication algorithms, security keychains, and level of authentication that can be configured:

- BFD Authentication Algorithms on page 793
- Security Authentication Keychains on page 793
- Strict Versus Loose Authentication on page 794

BFD Authentication Algorithms

JUNOS Software supports the following algorithms for BFD authentication:

- simple-password—Plain-text password. One to 16 bytes of plain text are used to authenticate the BFD session. One or more passwords may be configured. This method is the least secure and should be used only when BFD sessions are not subject to packet interception.
- keyed-md5—Keyed Message Digest 5 hash algorithm for sessions with transmit and receive intervals greater than 100 ms. To authenticate the BFD session, keyed MD5 uses one or more secret keys (generated by the algorithm) and a sequence number that is updated periodically. With this method, packets are accepted at the receiving end of the session if one of the keys matches and the sequence number is greater than or equal to the last sequence number received. Although more secure than a simple password, this method is vulnerable to replay attacks. Increasing the rate at which the sequence number is updated can reduce this risk.
- meticulous-keyed-md5—Meticulous keyed Message Digest 5 hash algorithm. This method works in the same manner as keyed MD5, but the sequence number is updated with every packet. Although more secure than keyed MD5 and simple passwords, this method may take additional time to authenticate the session.
- keyed-sha-1—Keyed Secure Hash Algorithm I for sessions with transmit and receive intervals greater than 100 ms. To authenticate the BFD session, keyed SHA uses one or more secret keys (generated by the algorithm) and a sequence number that is updated periodically. The key is not carried within the packets. With this method, packets are accepted at the receiving end of the session if one of the keys matches and the sequence number is greater than the last sequence number received.
- meticulous-keyed-sha-1—Meticulous keyed Secure Hash Algorithm I. This method works in the same manner as keyed SHA, but the sequence number is updated with every packet. Although more secure than keyed SHA and simple passwords, this method may take additional time to authenticate the session.

NOTE: Nonstop active routing (NSR) is not supported with meticulous-keyed-md5 and meticulous-keyed-sha-1 authentication algorithms. BFD sessions using these algorithms may go down after a switchover.

Security Authentication Keychains

The security authentication keychain defines the authentication attributes used for authentication key updates. When the security authentication keychain is configured

and associated with a protocol through the keychain name, authentication key updates can occur without interrupting routing and signaling protocols.

The authentication keychain contains one or more keychains. Each keychain contains one or more keys. Each key holds the secret data and the time at which the key becomes valid. The algorithm and keychain must be configured on both ends of the BFD session, and they must match. Any mismatch in configuration prevents the BFD session from being created.

BFD allows multiple clients per session, and each client can have its own keychain and algorithm defined. To avoid confusion, we recommend specifying only one security authentication keychain.

Strict Versus Loose Authentication

By default, strict authentication is enabled and authentication is checked at both ends of each BFD session. Optionally, to smooth migration from nonauthenticated sessions to authenticated sessions, you can configure *loose checking*. When loose checking is configured, packets are accepted without authentication being checked at each end of the session. This feature is intended for transitional periods only.

Related Topics

- Configuring BFD Authentication for BGP on page 794
- bfd-liveness-detection statement
- authentication-key-chains statement in the *JUNOS System Basics Configuration Guide*
- show bfd session command in the *JUNOS Routing Protocols and Policies Command Reference*
- Configuring BFD for BGP on page 789

Configuring BFD Authentication for BGP

Beginning with JUNOS Release 9.6, you can configure authentication for BFD sessions running over BGP. Only three steps are needed to configure authentication on a BFD session:

1. Specify the BFD authentication algorithm for the BGP protocol.
2. Associate the authentication keychain with the BGP protocol.
3. Configure the related security authentication keychain.

The following sections provide instructions for configuring and viewing BFD authentication on BGP:

- Configuring BFD Authentication Parameters on page 795
- Viewing Authentication Information for BFD Sessions on page 796

Configuring BFD Authentication Parameters

BFD authentication can be configured for the entire BGP protocol, or a specific BGP group, neighbor, or routing instance.

To configure BFD authentication:

1. Specify the algorithm (keyed-md5, keyed-sha-1, meticulous-keyed-md5, meticulous-keyed-sha-1, or simple-password) to use.

```
[edit]
user@host# set protocols bgp bfd-liveness-detection authentication algorithm
   keyed-sha-1
user@host# set protocols bgp group bgp-gr1 bfd-liveness-detection authentication
   algorithm keyed-sha-1
user@host# set protocols bgp group bgp-gr1 neighbor 10.10.10.7
   bfd-liveness-detection authentication algorithm keyed-sha-1
```

NOTE: Nonstop active routing (NSR) is not supported with meticulous-keyed-md5 and meticulous-keyed-sha-1 authentication algorithms. BFD sessions using these algortihms may go down after a switchover.

2. Specify the keychain to be used to associate BFD sessions on BGP with the unique security authentication keychain attributes. The keychain name you specify must match a keychain name configured at the [edit security authentication key-chains] hierarchy level.

```
[edit]
user@host# set protocols bgp bfd-liveness-detection authentication keychain
   bfd-bgp
user@host# set protocols bgp group bgp-gr1 bfd-liveness-detection authentication
   keychain bfd-bgp
user@host# set protocols bgp group bgp-gr1 neighbor 10.10.10.7
   bfd-liveness-detection authentication keychain bfd-bgp
```

NOTE: The algorithm and keychain must be configured on both ends of the BFD session, and they must match. Any mismatch in configuration prevents the BFD session from being created.

3. Specify the unique security authentication information for BFD sessions:
   - The matching *key-chain-name* as specified in step 2.
   - At least one *key*, a unique integer between 0 and 63. Creating multiple keys allows multiple clients to use the BFD session.
   - The *secret-data* used to allow access to the session.
   - The time at which the authentication key becomes active, *yyyy-mm-dd.hh:mm:ss*.

```
[edit security]
user@host# authentication-key-chains key-chain bfd-bgp key 53 secret
   $9$ggaJDmPQ6/tJgF/AtREVsyPsnCtUHm start-time 2009-06-14.10:00:00
```

4. (Optional) Specify loose authentication checking if you are transitioning from nonauthenticated sessions to authenticated sessions.

```
[edit]
user@host> set protocols bgp bfd-liveness-detection authentication loose-check
user@host> set protocols bgp group bgp-gr1 bfd-liveness-detection authentication
   loose-check
user@host> set protocols bgp group bgp-gr1 neighbor 10.10.10.7
   bfd-liveness-detection authentication loose-check
```

5. (Optional) View your configuration using the show bfd session detail or show bfd session extensive command.

6. Repeat these steps to configure the other end of the BFD session.

NOTE: BFD authentication is only supported in the domestic image and is not available in the export image.

Viewing Authentication Information for BFD Sessions

You can view the existing BFD authentication configuration using the show bfd session detail and show bfd session extensive commands.

The following example shows BFD authentication configured for the bgp-gr1 BGP group. It specifies the keyed SHA-1 authentication algorithm and a keychain name of bfd-bgp. The authentication keychain is configured with two keys. Key 1 contains the secret data "9ggaJDmPQ6/tJgF/AtREVsyPsnCtUHm" and a start time of June 1, 2009, at 9:46:02 AM PST. Key 2 contains the secret data "9a5jiKW9I.reP38ny.TszF2/9" and a start time of June 1, 2009, at 3:29:20 PM PST.

```
[edit protocols bgp]
group bgp-gr1 {
   bfd-liveness-detection {
      authentication {
         algorithm keyed-sha-1;
         key-chain bfd-bgp;
      }
   }
}
[edit security]
authentication key-chains {
   key-chain bfd-bgp {
      key 1 {
         secret "$9$ggaJDmPQ6/tJgF/AtREVsyPsnCtUHm";
         start-time "2009-6-1.09:46:02 -0700";
      }
      key 2 {
         secret "$9$a5jiKW9I.reP38ny.TszF2/9";
```

```
            start-time "2009-6-1.15:29:20 -0700";
        }
    }
}
```

If you commit these updates to your configuration, you would see output similar to the following. In the output for the show bfd sessions detail command, Authenticate is displayed to indicate that BFD authentication is configured. For more information about the configuration, use the show bfd sessions extensive command. The output for this command provides the keychain name, the authentication algorithm and mode for each client in the session, and the overall BFD authentication configuration status, keychain name, and authentication algorithm and mode.

show bfd sessions detail

```
user@host# show bfd session detail

                                                Detect   Transmit
Address                 State     Interface     Time     Interval  Multiplier
50.0.0.2                Up        ge-0/1/5.0    0.900     0.300        3
 Client BGP, TX interval 0.300, RX interval 0.300, Authenticate
 Session up time 3d 00:34
 Local diagnostic None, remote diagnostic NbrSignal
 Remote state Up, version 1
 Replicated
```

show bfd sessions extensive

```
user@host# show bfd session extensive
                                                Detect   Transmit
Address                 State     Interface     Time     Interval  Multiplier
50.0.0.2                Up        ge-0/1/5.0    0.900     0.300        3
 Client BGP, TX interval 0.300, RX interval 0.300, Authenticate
        keychain bfd-bgp, algo keyed-sha-1, mode strict
 Session up time 00:04:42
 Local diagnostic None, remote diagnostic NbrSignal
 Remote state Up, version 1
 Replicated
 Min async interval 0.300, min slow interval 1.000
 Adaptive async TX interval 0.300, RX interval 0.300
 Local min TX interval 0.300, minimum RX interval 0.300, multiplier 3
 Remote min TX interval 0.300, min RX interval 0.300, multiplier 3
 Local discriminator 2, remote discriminator 2
 Echo mode disabled/inactive
 Authentication enabled/active, keychain bfd-bgp, algo keyed-sha-1, mode strict
```

Related Topics

- Overview of BFD Authentication for BGP on page 792
- bfd-liveness-detection
- Configuring BFD for BGP on page 789
- authentication-key-chains statement in the *JUNOS System Basics Configuration Guide*
- show bfd session command in the *JUNOS Routing Protocols and Policies Command Reference*

Limiting TCP Segment Size for BGP

TCP path MTU discovery helps avoid BGP packet fragmentation. However, enabling TCP path MTU discovery creates ICMP vulnerability. To prevent these ICMP vulnerability issues, you can configure the TCP maximum segment size (MSS) globally, or for each BGP peer. You can also configure the advertised MSS value for each BGP peer to prevent fragmentation of packets sent by the BGP peer.

To configure the TCP MSS value, include the tcp-mss statement:

```
tcp-mss segment-size;
```

For a list of hierarchy levels at which you can include this statement, see the statement summary section for this statement.

Include the tcp-mss statement for a specific BGP neighbor to send the specified segment size to the BGP neighbor as the advertised MSS. The configured MSS value is also used as the maximum segment size for the sender. If the MSS value from the BGP neighbor is less than the MSS value configured, the MSS value from the BGP neighbor is used as the maximum segment size for the sender.

Configuring the BGP Monitoring Protocol

The BGP Monitoring Protocol enables you to collect data from the BGP Adjacency-RIB-In routing tables and to have that data sent periodically to a monitoring station. The JUNOS Software implementation of the BGP Monitoring Protocol (BMP) is based on Internet draft draft-scudder-bmp-01.txt, *BGP Monitoring Protocol*.

To configure the BGP Monitoring Protocol, include the bmp statement at the [edit routing-options] hierarchy level:

```
[edit routing-options]
bmp {
    <memory-limit bytes>;
    station-address (ip-address | name);
    station-port port-number;
    <statistics-timeout seconds>;
}
```

To configure the monitoring station to which BMP data is sent, you must configure both the station-address and station-port statements. For the station address, you can specify either the IP address or the name of the monitoring station. For *name*, specify a valid URL.

You can optionally specify how often to send data to the monitoring station. The default is 1 hour. To modify this interval, include the statistics-timeout *seconds* statement. For *seconds*, you can specify a value from 15 through 65,535. By default, the routing device stops collecting BMP data when it exceeds a threshold of 10 MB. You can modify the value of this threshold by including the memory-limit *bytes* statement. For *bytes*, specify a value from 1048576 to 52428800. If the routing

device stops collecting BMP data after exceeding the configured memory threshold, the router waits 10 minutes before attempting to resume the BMP session.

Tracing BGP Protocol Traffic

You can trace various BGP protocol traffic to help you debug the BGP protocol issues. To trace BGP protocol traffic include the **traceoptions** statement at the [**edit protocols bgp**] hierarchy level. For routing instances, include the **traceoptions** statement at the [**edit routing-instances** *routing-instance-name* **protocols bgp**] hierarchy level.

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

You can specify the following BGP protocol-specific trace options using the **flag** statement:

- **4byte-as**—4-byte AS events
- **bfd**—BFD protocol events
- **damping**—Damping operations
- **graceful-restart**—Graceful restart events
- **keepalive**—BGP keepalive messages
- **nsr-synchronization**—Nonstop active routing synchronization events
- **open**—BGP open packets. These packets are sent between peers when they are establishing a connection.
- **packets**—All BGP protocol packets
- **refresh**—BGP refresh packets
- **update**—BGP update packets. These packets provide routing updates to BGP systems.

Global tracing options are inherited from the configuration set by the **traceoptions** statement at the [**edit routing-options**] hierarchy level. You can override the following global trace options for the BGP protocol using the **traceoptions flag** statement included at the [**edit protocols bgp**] hierarchy level:

- **all**—All tracing operations
- **general**—All normal operations and routing table changes (a combination of the normal and route trace operations)
- **normal**—Normal events
- **policy**—Policy processing
- **route**—Routing information
- **state**—State transitions

- task—Routing protocol task processing
- timer—Routing protocol timer processing

You can optionally specify one or more of the following flag modifiers:

- detail—Detailed trace information
- filter—Filter trace information. Applies only to route and damping tracing flags.
- receive—Packets being received
- send—Packets being transmitted

NOTE: Use the all trace flag and the detail flag modifier with caution as these may cause the CPU to become very busy.

NOTE: If you only enable the update flag, received keepalive messages do not generate a trace message.

You can filter trace statements and output only the statement information that passes through the filter by specifying the filter flag modifier. The filter modifier is only supported for the route and damping tracing flags.

The match-on statement specifies filter matches based on prefixes. It is used to match on route filters.

NOTE: Per-neighbor trace filtering is not supported on a BGP per-neighbor level for route and damping flags. Trace option filtering support is on a peer group level.

Examples: Tracing BGP Protocol Traffic

Trace only unusual or abnormal operations to routing-log, and trace detailed information about all BGP messages to bgp-log:

```
[edit]
routing-options {
   traceoptions {
      file routing-log;
   }
   autonomous-system 23;
}
protocols {
   bgp {
      group 23 {
         type external;
         peer-as 56;
         traceoptions {
            file bgp-log size 10k files 5;
```

```
            flag packets detail;
        }
        0.0.0.0/0;
    }
  }
}
```

Trace only update messages received from the configured peer:

```
[edit]
routing-options {
  autonomous-system 23;
  router-id 10.0.0.1;
}
protocols {
  bgp {
    group 23 {
      type external;
      peer-as 56;
      neighbor boojum.snark.net {
        traceoptions {
          file bgp-log size 10k files 2;
          flag update detail;
        }
      }
    }
  }
}
```

Trace only messages that pass the policy based on prefix match:

```
[edit]
protocols {
  bgp {
    traceoptions {
      file bgp-tr size 5m files 10;
      flag route filter policy couple-route match-on prefix;
    }
  }
}
```

Related Topics

- traceoptions statement
- For general information about tracing, see the tracing and logging information in the *JUNOS System Basics Configuration Guide*.

Chapter 36

Summary of BGP Configuration Statements

The following sections explain each of the BGP configuration statements. The statements are organized alphabetically.

accept-remote-nexthop

Syntax accept-remote-nexthop;

Hierarchy Level
[edit logical-systems *logical-system-name* protocols bgp],
[edit logical-systems *logical-system-name* protocols bgp group *group-name*],
[edit logical-systems *logical-system-name* protocols bgp group *group-name* neighbor *address*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address*],
[edit protocols bgp],
[edit protocols bgp group *group-name*],
[edit protocols bgp group *group-name* neighbor *address*],
[edit routing-instances *routing-instance-name* protocols bgp],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name*],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address*]

Release Information Statement introduced in JUNOS Release 8.5.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Specify that a single-hop EBGP peer accept a remote next hop with which it does not share a common subnet. Configure a separate import policy on the EBGP peer to specify the remote next hop. You cannot configure the multihop statement at the same time.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- multipath
- Configuring Single-Hop EBGP Peers to Accept Remote Next Hops on page 750
- Applying Policies to BGP Routes on page 781

accepted-prefix-limit

Syntax

```
accepted-prefix-limit {
    maximum number;
    teardown <percentage-threshold> idle-timeout (forever | minutes);
}
```

Hierarchy Level

[edit logical-systems *logical-system-name* protocols bgp family (inet | inet6) (any | flow | labeled-unicast | multicast | unicast)],
[edit logical-systems *logical-system-name* protocols bgp family route-target],
[edit logical-systems *logical-system-name* protocols bgp group *group-name* family (inet | inet6) (any | flow | labeled-unicast | multicast | unicast)],
[edit logical-systems *logical-system-name* protocols bgp group *group-name* family route-target],
[edit logical-systems *logical-system-name* protocols bgp group *group-name* neighbor *address* family (inet | inet6) (any | flow | labeled-unicast | multicast | unicast)],
[edit logical-systems *logical-system-name* protocols bgp group *group-name* neighbor *address* family route-target],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp family (inet | inet6) (any | flow | labeled-unicast | multicast | unicast)],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp family route-target],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name* family (inet | inet6) (any | flow | labeled-unicast | multicast | unicast)],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name* family route-target],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address* family (inet | inet6) (any | flow | labeled-unicast | multicast | unicast)],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address* family route-target],
[edit protocols bgp family (inet | inet6) (any | flow | labeled-unicast | multicast | unicast)],
[edit protocols bgp family route-target],
[edit protocols bgp group *group-name* family (inet | inet6) (any | flow | labeled-unicast | multicast | unicast)],
[edit protocols bgp group *group-name* family route-target],
[edit protocols bgp group *group-name* neighbor *address* family (inet | inet6) (any | flow | labeled-unicast | multicast | unicast)],
[edit protocols bgp group *group-name* neighbor *address* family route-target],
[edit routing-instances *routing-instance-name* protocols bgp family (inet | inet6) (any | flow | labeled-unicast | multicast | unicast)],
[edit routing-instances *routing-instance-name* protocols bgp family route-target],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name* family (inet | inet6) (any | flow | labeled-unicast | multicast | unicast)],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name* family route-target],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address* family (inet | inet6) (any | flow | labeled-unicast | multicast | unicast)],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address* family route-target]

Release Information Statement introduced in JUNOS Release 9.2.
Statement introduced in JUNOS Release 9.2 for EX Series switches.

Description Configure a limit to the number of prefixes that can be accepted on a BGP peering session. When that limit is exceeded, a system log message is sent. You can optionally specify to reset the BGP session when the number of accepted prefixes exceeds the specified limit.

Options idle-timeout (forever | *minutes*)—Specify that a BGP session that has been reset is not reestablished until after the specified timeout period. Specify forever to prevent the BGP session from being reestablished until the clear bgp neighbor command is issued.

maximum *number*—Limit the number of prefixes that can be accepted on a BGP peering session. A system log message is sent when that number is exceeded.
Range: 1 through 4,294,967,295 ($2^{32} - 1$)

teardown <*percentage* 1/n threshold>—Specify to reset the BGP peering session when the specified limit to the number of prefixes that can be accepted is exceeded. If you specify a percentage, a system log message is sent when the accepted number of prefixes on the BGP session exceeds the specified percentage of the configured limit. After a BGP session is reset, it is reestablished within a short time unless you include the idle-timeout statement.
Range: 1 through 100
Range: 1 through 2400

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- prefix-limit
- Limiting the Number of Prefixes Accepted on a BGP Peering Session on page 771

advertise-external

Syntax

```
advertise-external {
    conditional;
}
```

Hierarchy Level

[edit logical-systems *logical-system-name* protocols bgp group *group-name*],
[edit logical-systems *logical-system-name* protocols bgp group *group-name* neighbor *address*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address*],
[edit protocols bgp group *group-name*],
[edit protocols bgp group *group-name* neighbor *address*],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name*],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *neighbor-address*]

Release Information

Statement introduced in JUNOS Release 9.3.
Statement introduced in JUNOS Release 9.3 for EX Series switches.

Description

Have BGP advertise the best external route into an IBGP mesh group, a route reflector cluster, or an AS confederation even if the best route is an internal route.

Options

conditonal—(Optional) Advertise the best external path only if the route selection process reaches the point where the multiple exit discriminator (MED) metric is evaluated. As a result, an external path with an AS path worse than that of the active path is not advertised.

Required Privilege Level

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

Related Topics

- advertise-inactive
- Configuring BGP to Advertise the Best External Route to Internal Peers on page 783

advertise-inactive

Syntax advertise-inactive;

Hierarchy Level
[edit logical-systems *logical-system-name* protocols bgp],
[edit logical-systems *logical-system-name* protocols bgp group *group-name*],
[edit logical-systems *logical-system-name* protocols bgp group *group-name* neighbor *address*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address*],
[edit protocols bgp],
[edit protocols bgp group *group-name*],
[edit protocols bgp group *group-name* neighbor *address*],
[edit routing-instances *routing-instance-name* protocols bgp],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name*],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address*]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Have BGP advertise the best route even if the routing table did not select it to be an active route.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Setting BGP to Advertise Inactive Routes on page 783

advertise-peer-as

Syntax advertise-peer-as;

Hierarchy Level
[edit logical-systems *logical-system-name* protocols bgp],
[edit logical-systems *logical-system-name* protocols bgp group *group-name*],
[edit logical-systems *logical-system-name* protocols bgp group *group-name* neighbor *address*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address*],
[edit protocols bgp],
[edit protocols bgp group *group-name*],
[edit protocols bgp group *group-name* neighbor *address*],
[edit routing-instances *routing-instance-name* protocols bgp],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name*],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address*]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Disable the default behavior of suppressing AS routes.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Disabling Suppression of Route Advertisements on page 785

aggregate-label

Syntax

```
aggregate-label {
    community community-name;
}
```

Hierarchy Level

[edit logical-systems *logical-system-name* protocols bgp family inet labeled-unicast],
[edit logical-systems *logical-system-name* protocols bgp family inet-vpn labeled-unicast],
[edit protocols bgp family inet labeled-unicast],
[edit protocols bgp family inet-vpn labeled-unicast],
[edit protocols bgp family inet6 labeled-unicast]

Release Information

Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description

Enable aggregate labels for VPN traffic.

Options

community *community-name*—Specify the name of the community to which to apply the aggregate label.

Required Privilege Level

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

Related Topics

- Configuring Aggregate Labels for VPNs on page 743

allow

Syntax allow (all | [*network/mask-length*]);

Hierarchy Level [edit logical-systems *logical-system-name* protocols bgp group *group-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name*],
[edit protocols bgp group *group-name*],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name*]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Implicitly configure BGP peers, allowing peer connections from any of the specified networks or hosts. To configure multiple BGP peers, configure one or more networks and hosts within a single allow statement or include multiple allow statements.

Options all—Allow all addresses, which is equivalent to 0.0.0.0/0 (or ::/0).

network/mask-length—IPv6 or IPv4 network number of a single address or a range of allowable addresses for BGP peers, followed by the number of significant bits in the subnet mask.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics

- neighbor
- Minimum BGP Configuration on page 723
- Configuring BGP Groups and Peers on page 728

as-override

Syntax as-override;

Hierarchy Level
[edit logical-systems *logical-system-name* protocols bgp group *group-name*],
[edit logical-systems *logical-system-name* protocols bgp group *group-name* neighbor *address*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address*],
[edit protocols bgp group *group-name*],
[edit protocols bgp group *group-name* neighbor *address*],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name*],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address*]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Compare the AS path of an incoming advertised route with the AS number of the BGP peer under the group and replace all occurrences of the peer AS number in the AS path with its own AS number before advertising the route to the peer.

NOTE: The as-override statement is specific to a particular BGP group. This statement does not affect peers from the same remote AS configured in different groups.

Enabling the AS override feature allows routes originating from an AS to be accepted by a router residing in the same AS. Without AS override enabled, the routing device refuses the route advertisement once the AS path shows that the route originated from its own AS. This is done by default to prevent route loops. The as-override statement overrides this default behavior.

Note that enabling the AS override feature may result in routing loops. Use this feature only for specific applications that require this type of behavior, and in situations with strict network control. One application is the IGP protocol between the provider edge routing device and the customer edge routing device in a virtual private network. For more information, see the *JUNOS MPLS Applications Configuration Guide*.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring BGP Groups and Peers on page 728

authentication-algorithm

Syntax authentication-algorithm *algorithm*;

Hierarchy Level
[edit logical-systems *logical-system-name* protocols bgp],
[edit logical-systems *logical-system-name* protocols bgp group *group-name*],
[edit logical-systems *logical-system-name* protocols bgp group *group-name* neighbor *address*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address*],
[edit protocols bgp],
[edit protocols bgp group *group-name*],
[edit protocols bgp group *group-name* neighbor *address*],
[edit routing-instances *routing-instance-name* protocols bgp],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name*],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address*]

Release Information Statement introduced in JUNOS Release 8.0.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Configure an MD5 authentication algorithm type.

Options *algorithm*—Type of authentication algorithm. Specify either md5 or hmac-sha-1-96 as the algorithm type.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring Authentication for BGP on page 743

authentication-key

Syntax authentication-key *key*;

Hierarchy Level
[edit logical-systems *logical-system-name* protocols bgp],
[edit logical-systems *logical-system-name* protocols bgp group *group-name*],
[edit logical-systems *logical-system-name* protocols bgp group *group-name* neighbor *address*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address*],
[edit protocols bgp],
[edit protocols bgp group *group-name*],
[edit protocols bgp group *group-name* neighbor *address*],
[edit routing-instances *routing-instance-name* protocols bgp],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name*],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address*]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Configure an MD5 authentication key (password). Neighboring routing devices use the same password to verify the authenticity of BGP packets sent from this system.

Options *key*—Authentication password. It can be up to 126 characters. Characters can include any ASCII strings. If you include spaces, enclose all characters in quotation marks (“ ”).

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring Authentication for BGP on page 743

authentication-key-chain

Syntax authentication-key-chain *key-chain*;

Hierarchy Level
[edit logical-systems *logical-system-name* protocols bgp],
[edit logical-systems *logical-system-name* protocols bgp group *group-name*],
[edit logical-systems *logical-system-name* protocols bgp group *group-name* neighbor *address*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address*],
[edit protocols bgp],
[edit protocols bgp group *group-name*],
[edit protocols bgp group *group-name* neighbor *address*],
[edit routing-instances *routing-instance-name* protocols bgp],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name*],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address*]

Release Information Statement introduced in JUNOS Release 8.0.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Apply and enable an authentication keychain to the routing device.

Options *key-chain*—Authentication keychain name. It can be up to 126 characters. Characters can include any ASCII strings. If you include spaces, enclose all characters in quotation marks (“ ”).

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring Authentication for BGP on page 743

bfd-liveness-detection

Syntax

```
bfd-liveness-detection {
    authentication {
        algorithm algorithm-name;
        key-chain key-chain-name;
        <loose-check>;
    }
    detection-time {
        threshold milliseconds;
    }
    holddown-interval milliseconds;
    minimum-interval milliseconds;
    minimum-receive-interval milliseconds;
    multiplier number;
    no-adaptation;
    transmit-interval {
        threshold milliseconds;
        minimum-interval milliseconds;
    }
    version (1 | automatic);
}
```

Hierarchy Level

[edit logical-systems *logical-system-name* protocols bgp],
[edit logical-systems *logical-system-name* protocols bgp group *group-name*],
[edit logical-systems *logical-system-name* protocols bgp group *group-name* neighbor *address*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address*],
[edit protocols bgp],
[edit protocols bgp group *group-name*],
[edit protocols bgp group *group-name* neighbor *address*],
[edit routing-instances *routing-instance-name* protocols bgp],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name*],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address*]

Release Information

Statement introduced in JUNOS Release 8.1.
Statement introduced in JUNOS Release 9.0 for EX Series switches.
detection-time threshold and transmit-interval threshold options introduced in JUNOS Release 8.2
Support for logical routers introduced in JUNOS Release 8.3.
Support for IBGP and multihop EBGP sessions introduced in JUNOS Release 8.3.
holddown-interval statement introduced in JUNOS Release 8.5. You can configure this statement only for EBGP peers at the [edit protocols bgp group *group-name* neighbor *address*] hierarchy level.
no-adaptation statement introduced in JUNOS Release 9.0.

Support for BFD authentication introduced in JUNOS Release 9.6.

Description Configure bidirectional failure detection timers and authentication.

For IBGP and multihop EBGP support, configure the bfd-liveness-detection statement at the global [edit bgp protocols] hierarchy level. You can also configure IBGP and multihop support for a routing instance or a logical system.

Options authentication algorithm *algorithm-name* —Configure the algorithm used to authenticate the specified BFD session: simple-password, keyed-md5, keyed-sha-1, meticulous-keyed-md5, meticulous-keyed-sha-1.

authentication key-chain *key-chain-name*—Associate a security key with the specified BFD session using the name of the security keychain. The keychain name must match one of the keychains configured in the authentication-key-chains key-chain statement at the [edit security] hierarchy level.

authentication loose-check—(Optional) Configure loose authentication checking on the BFD session. Use only for transitional periods when authentication may not be configured at both ends of the BFD session.

detection-time threshold *milliseconds*—Configure a threshold. When the BFD session detection time adapts to a value equal to or greater than the threshold, a single trap and a single system log message are sent.

holddown-interval *milliseconds*—Configure an interval specifying how long a BFD session must remain up before a state change notification is sent.
Range: 0 through 255,000
Default: 0

NOTE: You can configure the holddown-interval option only for EBGP peers.

minimum-interval *milliseconds*—Configure the minimum intervals at which the local routing device transmits hello packets and then expects to receive a reply from a neighbor with which it has established a BFD session.
Range: 1 through 255,000

minimum-receive-interval *milliseconds*— Configure only the minimum interval at which the local routing device expects to receive a reply from a neighbor with which it has established a BFD session.
Range: 1 through 255,000

multiplier *number*—Configure the number of hello packets not received by a neighbor that causes the originating interface to be declared down.
Range: 1 through 255
Default: 3

no-adaptation—Configure BFD sessions not to adapt to changing network conditions. We recommend that you not disable BFD adaptation unless it is preferable to not to have BFD adaptation enabled in your network.

transmit-interval threshold *milliseconds*—Configure a threshold. When the BFD session transmit interval adapts to a value greater than the threshold, a single trap and a single system message are sent. The interval threshold must be greater than the minimum transmit interval.
Range: 0 through 4,294,967,295 ($2^{32} - 1$)

transmit-interval minimum-interval *milliseconds*—Configure only the minimum interval at which the local routing device transmits hello packets to a neighbor with which it has established a BFD session.
Range: 1 through 255,000

version—Configure the BFD version to detect.
Range: 1 or automatic (autodetect the BFD version)
Default: automatic

The remaining statements are explained separately.

Required Privilege Level

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

Related Topics

- Configuring BFD for BGP on page 789
- Configuring BFD Authentication for BGP on page 794

bgp

Syntax bgp { ... }

Hierarchy Level [edit logical-systems *logical-system-name* protocols bgp],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp],
[edit protocols],
[edit routing-instances *routing-instance-name* protocols]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Enable BGP on the routing device or for a routing instance.

Default BGP is disabled.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Enabling BGP on page 724

bgp-orf-cisco-mode

Syntax bgp-orf-cisco-mode;

Hierarchy Level
[edit logical-systems *logical-system-name* protocols bgp outbound-route-filter],
[edit logical-systems *logical-system-name* protocols bgp group *group-name* outbound-route-filter],
[edit logical-systems *logical-system-name* protocols bgp group *group-name* neighbor *address* outbound-route-filter],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp outbound-route-filter],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name* outbound-route-filter,
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address* outbound-route-filter],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* routing-options outbound-route-filter],
[edit logical-systems *logical-system-name* routing-options outbound-route-filter],
[edit protocols bgp outbound-route-filter],
[edit protocols bgp group *group-name* outbound-route-filter],
[edit protocols bgp group *group-name* neighbor *address* outbound-route-filter],
[edit routing-instances *routing-instance-name* protocols bgp outbound-route-filter],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name* outbound-route-filter],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address* outbound-route-filter],
[edit routing-instances *routing-instance-name* routing-options outbound-route-filter],
[edit routing-options outbound-route-filter]

Release Information
Statement introduced in JUNOS Release 9.2.
Statement introduced in JUNOS Release 9.2 for EX Series switches.
Support for the BGP group and neighbor hierarchy levels introduced in JUNOS Release 9.2.
Support for the BGP group and neighbor hierarchy levels introduced in JUNOS Release 9.3 for EX Series switches.

Description Enable interoperability with routing devices that use the vendor-specific outbound route filter compatibility code of 130 and code type of 128.

NOTE: To enable interoperability for all BGP peers configured on the routing device, include the statement at the [edit routing-options outbound-route-filter] hierarchy level.

Default Disabled

Required Privilege Level
routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Applying Filters Provided by BGP Peers to Outbound Routes on page 780

bmp

Syntax

```
bmp {
    memory limit bytes;
    station-address (ip-address | name);
    station-port port-number;
    statistics-timeout seconds;
}
```

Hierarchy Level [edit routing-options]

Release Information Statement introduced in JUNOS Release 9.5.
Statement introduced in JUNOS Release 9.5 for EX Series switches.

Description Configure the BGP Monitoring Protocol (BMP), which enables the routing device to collect data from the BGP Adjacency-RIB-In routing tables and periodically send that data to a monitoring station.

Options

memory-limit *bytes*—(Optional) Specify a threshold at which to stop collecting BMP data if the limit is exceeded.
Default: 10 MB
Range: 1,048,576 through 52,428,800

station-address (*ip-address* | *name*)—Specify the IP address or a valid URL for the monitoring where BMP data should be sent.

station-port *port-number*—Specify the port number of the monitoring station to use when sending BMP data.

statistics-timeout *seconds*—(Optional) Specify how often to send BMP data to the monitoring station.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics

- Configuring the BGP Monitoring Protocol on page 798

cluster

Syntax cluster *cluster-identifier*;

Hierarchy Level
[edit logical-systems *logical-system-name* protocols bgp],
[edit logical-systems *logical-system-name* protocols bgp group *group-name*],
[edit logical-systems *logical-system-name* protocols bgp group *group-name* neighbor *address*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address*],
[edit protocols bgp],
[edit protocols bgp group *group-name*],
[edit protocols bgp group *group-name* neighbor *address*],
[edit routing-instances *routing-instance-name* protocols bgp],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name*],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address*]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Specify the cluster identifier to be used by the route reflector cluster in an internal BGP group.

Options *cluster-identifier*—IPv6 or IPv4 address to use as the cluster identifier.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- no-client-reflect
- Configuring BGP Route Reflection on page 761

damping

Syntax damping;

Hierarchy Level [edit logical-systems *logical-system-name* protocols bgp],
[edit logical-systems *logical-system-name* protocols bgp group *group-name*],
[edit logical-systems *logical-system-name* protocols bgp group *group-name* neighbor *address*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address*],
[edit protocols bgp],
[edit protocols bgp group *group-name*],
[edit protocols bgp group *group-name* neighbor *address*],
[edit routing-instances *routing-instance-name* protocols bgp],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name*],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address*]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Enable route flap damping.

Default Flap damping is disabled on the routing device.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring Flap Damping for BGP Routes on page 767
- *JUNOS Policy Framework Configuration Guide*

description

Syntax description *text-description*;

Hierarchy Level
[edit logical-systems *logical-system-name* protocols bgp],
[edit logical-systems *logical-system-name* protocols bgp group *group-name*],
[edit logical-systems *logical-system-name* protocols bgp group *group-name* neighbor *address*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address*],
[edit protocols bgp],
[edit protocols bgp group *group-name*],
[edit protocols bgp group *group-name* neighbor *address*],
[edit routing-instances *routing-instance-name* protocols bgp],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name*],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address*]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Text description of the global, group, or neighbor configuration.

Options *text-description*—Text description of the configuration. It is limited to 126 characters.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Defining BGP Global Properties on page 725
- Defining Group Properties on page 732
- Defining Peer Properties on page 734

disable

Syntax disable;

Hierarchy Level [edit logical-systems *logical-system-name* protocols bgp],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp],
[edit protocols bgp],
[edit routing-instances *routing-instance-name* protocols bgp]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Disable BGP on the system.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Defining BGP Global Properties on page 725

explicit-null

Syntax explicit-null;

Hierarchy Level
[edit logical-systems *logical-system-name* protocols mpls],
[edit logical-systems *logical-system-name* protocols bgp family inet labeled-unicast],
[edit logical-systems *logical-system-name* protocols bgp family inet6 labeled-unicast],
[edit logical-systems *logical-system-name* protocols bgp group *group-name* family inet labeled-unicast],
[edit logical-systems *logical-system-name* protocols bgp group *group-name* family inet6 labeled-unicast],
[edit logical-systems *logical-system-name* protocols bgp group *group-name* neighbor *address* family inet labeled-unicast],
[edit logical-systems *logical-system-name* protocols bgp group *group-name* neighbor *address* family inet6 labeled-unicast],
[edit logical-systems *logical-system-name* protocols ldp],
[edit logical-systems *logical-system-name* routing-instances *instance-name* protocols bgp family inet labeled-unicast],
[edit logical-systems *logical-system-name* routing-instances *instance-name* protocols bgp family inet6 labeled-unicast],
[edit logical-systems *logical-system-name* routing-instances *instance-name* protocols bgp group *group-name* family inet labeled-unicast],
[edit logical-systems *logical-system-name* routing-instances *instance-name* protocols bgp group *group-name* family inet6 labeled-unicast],
[edit logical-systems *logical-system-name* routing-instances *instance-name* protocols bgp group *group-name* neighbor *address* family inet labeled-unicast],
[edit logical-systems *logical-system-name* routing-instances *instance-name* protocols bgp group *group-name* neighbor *address* family inet6 labeled-unicast],
[edit logical-systems *logical-system-name* routing-instances *instance-name* protocols ldp],
[edit protocols mpls],
[edit protocols bgp family inet labeled-unicast],
[edit protocols bgp family inet6 labeled-unicast],
[edit protocols bgp group *group-name* family inet labeled-unicast],
[edit protocols bgp group *group-name* family inet6 labeled-unicast],
[edit protocols bgp group *group-name* neighbor *address* family inet labeled-unicast]
[edit protocols bgp group *group-name* neighbor *address* family inet6 labeled-unicast],
[edit protocols ldp],
[edit routing-instances *instance-name* protocols bgp family inet labeled-unicast],
[edit routing-instances *instance-name* protocols bgp family inet6 labeled-unicast],
[edit routing-instances *instance-name* protocols bgp group *group-name* family inet labeled-unicast],
[edit routing-instances *instance-name* protocols bgp group *group-name* family inet6 labeled-unicast],
[edit routing-instances *instance-name* protocols bgp group *group-name* neighbor *address* family inet labeled-unicast],
[edit routing-instances *instance-name* protocols bgp group *group-name* neighbor *address* family inet6 labeled-unicast],
[edit routing-instances *instance-name* protocols ldp]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Advertise label 0 to the egress routing device of an LSP.

Default If you do not include the explicit-null statement in the configuration, label 3 (implicit null) is advertised.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Advertising Explicit Null Labels to BGP Peers on page 742

export

Syntax
```
export [ policy-names ];
```

Hierarchy Level
```
[edit logical-systems logical-system-name protocols bgp],
[edit logical-systems logical-system-name protocols bgp group group-name],
[edit logical-systems logical-system-name protocols bgp group group-name
    neighbor address],
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols
    bgp],
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols
    bgp group group-name],
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols
    bgp group group-name neighbor address],
[edit protocols bgp],
[edit protocols bgp group group-name],
[edit protocols bgp group group-name neighbor address],
[edit routing-instances routing-instance-name protocols bgp],
[edit routing-instances routing-instance-name protocols bgp group group-name],
[edit routing-instances routing-instance-name protocols bgp group group-name
    neighbor address]
```

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Apply one or more policies to routes being exported from the routing table into BGP.

Options *policy-names*—Name of one or more policies.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- import
- Applying Policies to BGP Routes on page 781
- *JUNOS Policy Framework Configuration Guide*

family

Syntax

```
family {
    (inet | inet6 | inet-vpn | inet6-vpn | iso-vpn) {
        (any | flow | labeled-unicast | multicast | unicast) {
            accepted-prefix-limit {
                maximum number;
                teardown <percentage> <idle-timeout (forever | minutes)>;
            }
            <loops number>;
            prefix-limit {
                maximum number;
                teardown <percentage> <idle-timeout (forever | minutes)>;
            }
            rib-group group-name;
        }
        flow {
            no-validate policy-name;
        }
        labeled-unicast {
            accepted-prefix-limit {
                maximum number;
                teardown <percentage> <idle-timeout (forever | minutes)>;
            }
            aggregate-label {
                community community-name:
            }
            explicit-null {
                connected-only;
            }
            prefix-limit {
                maximum number;
                teardown <percentage> <idle-timeout (forever | minutes)>;
            }
            resolve-vpn;
            rib inet.3;
            rib-group group-name;
        }
    }
    route-target {
        accepted-prefix-limit {
            maximum number;
            teardown <percentage> <idle-timeout (forever | minutes)>;
        }
        advertise-default;
        external-paths number;
        prefix-limit {
            maximum number;
            teardown <percentage> <idle-timeout (forever | minutes)>;
        }
    }
    (inet-mdt | inet-mvpn | inet6-mvpn | l2-vpn) {
        signaling {
```

```
        accepted-prefix-limit {
            maximum number;
            teardown <percentage> <idle-timeout (forever | minutes)>;
        }
        <loops number>;
        prefix-limit {
            maximum number;
            teardown <percentage> <idle-timeout (forever | minutes)>;
            }
        rib-group group-name
      }
    }
}
```

Hierarchy Level

[edit logical-systems *logical-system-name* protocols bgp],
[edit logical-systems *logical-system-name* protocols bgp group *group-name*],
[edit logical-systems *logical-system-name* protocols bgp group *group-name* neighbor *address*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address*],
[edit protocols bgp],
[edit protocols bgp group *group-name*],
[edit protocols bgp group *group-name* neighbor *address*],
[edit routing-instances *routing-instance-name* protocols bgp],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name*],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address*]

Release Information

Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.
inet-mvpn and inet6-mpvn statements introduced in JUNOS Release 8.4.
inet-mdt statement introduced in JUNOS Release 9.4.
Support for the loops statement introduced in JUNOS Release 9.6.

Description

Enable multiprotocol BGP (MP-BGP) by configuring BGP to carry network layer reachability information (NLRI) for address families other than unicast IPv4, to specify MP-BGP to carry NLRI for the IPv6 address family, or to carry NLRI for VPNs.

Options any—Configure the family type to be both unicast and multicast.

inet—Configure NLRI parameters for IPv4.

inet6—Configure NLRI parameters for IPv6.

inet-mdt—Configure NLRI parameters for the multicast distribution tree (MDT) subaddress family identifier (SAFI) for IPv4 traffic in Layer 3 VPNs.

inet-mvpn—Configure NLRI parameters for IPv4 for multicast VPNs.

inet6-mvpn—Configure NLRI parameters for IPv6 for multicast VPNs.

inet-vpn—Configure NLRI parameters for IPv4 for Layer 3 VPNs.

inet6-vpn—Configure NLRI parameters for IPv6 for Layer 3 VPNs.

iso-vpn—Configure NLRI parameters for IS-IS for Layer 3 VPNs.

l2-vpn—Configure NLRI parameters for IPv4 for MPLS-based Layer 2 VPNs and VPLS.

labeled-unicast—Configure the family type to be labeled-unicast. This means that the BGP peers are being used only to carry the unicast routes that are being used by labeled-unicast for resolving the labeled-unicast routes. This statement is supported only with inet and inet6.

loops *number*—(Optional) Specify the maximum number of times that the AS number can appear in the AS path received from a BGP peer for the specified address family. For *number*, include a value from 1 through 10.

NOTE: When you configure the loops statement for a specific BGP address family, that value is used to evaluate the AS path for routes received by a BGP peer for the specified address family rather than the loops value configured for the global AS number.

multicast—Configure the family type to be multicast. This means that the BGP peers are being used only to carry the unicast routes that are being used by multicast for resolving the multicast routes.

unicast—Configure the family type to be unicast. This means that the BGP peers only carry the unicast routes that are being used for unicast forwarding purposes.
Default: unicast

The remaining statements are explained separately.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- autonomous-system
- local-as
- Enabling Multiprotocol BGP on page 767

flow

Syntax

```
flow {
    no-validate policy-name;
}
```

Hierarchy Level

[edit protocols bgp group *group-name* family (inet | inet-vpn)],
[edit protocols bgp group *group-name* neighbor *address* family (inet | inet-vpn)],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name* family (inet | inet-vpn)],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address* family (inet | inet-vpn)]

Release Information

Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description

Enables BGP to support flow routes.

NOTE: This statement is supported for the default instance, VRF instance, and virtual-router instance only. It is configured with the instance-type statement at the [edit routing-instance *instance-name*] hierarchy level. For VPNs, this statement is supported for the default instance only.

The remaining statements are explained separately.

Required Privilege Level

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

Related Topics

- Enabling BGP to Carry Flow-Specification Routes on page 773

graceful-restart

Syntax

```
graceful-restart {
    disable;
    restart-time seconds;
    stale-routes-time seconds;
}
```

Hierarchy Level

[edit logical-systems *logical-system-name* protocols bgp],
[edit logical-systems *logical-system-name* protocols bgp group *group-name*],
[edit logical-systems *logical-system-name* protocols bgp group *group-name* neighbor *address*],
[edit protocols bgp],
[edit protocols bgp group *group-name*],
[edit protocols bgp group *group-name* neighbor *address*]

Release Information

Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description

Configure graceful restart for BGP.

Options

disable—Disable graceful restart for BGP.

restart-time *seconds*—Time period when the restart is expected to be complete.
Range: 1 through 600 seconds

stale-routes-time *seconds*—Maximum time that stale routes are kept during restart.
Range: 1 through 600 seconds

Required Privilege Level

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

Related Topics

- Configuring Graceful Restart on page 124
- Configuring Graceful Restart for BGP on page 742
- *JUNOS High Availability Configuration Guide*

group

Syntax

```
group group-name {
    advertise-inactive;
    allow [ network/mask-length ];
    authentication-key key;
    cluster cluster-identifier;
    damping;
    description text-description;
    export [ policy-names ];
    family {
        (inet | inet6 | inet-vpn | inet6-vpn | l2-vpn) {
            (any | multicast | unicast | signaling) {
                accepted-prefix-limit {
                    maximum number;
                    teardown <percentage> <idle-timeout (forever | minutes)>;
                }
                prefix-limit {
                    maximum number;
                    teardown <percentage> <idle-timeout (forever | minutes)>;
                }
                rib-group group-name;
            }
            flow {
                no-validate policy-name;
            }
            labeled-unicast {
                accepted-prefix-limit {
                    maximum number;
                    teardown <percentage> <idle-timeout (forever | minutes)>;
                }
                explicit-null {
                    connected-only;
                }
                prefix-limit {
                    maximum number;
                    teardown <percentage> <idle-timeout (forever | minutes)>;
                }
                resolve-vpn;
                rib inet.3;
                rib-group group-name;
            }
        }
        route-target {
            accepted-prefix-limit {
                maximum number;
                teardown <percentage> <idle-timeout (forever | minutes)>;
            }
            advertise-default;
            external-paths number;
            prefix-limit {
                maximum number;
                teardown <percentage> <idle-timeout (forever | minutes)>;
```

```
        }
      }
    }
    hold-time seconds;
    import [ policy-names ];
    ipsec-sa ipsec-sa;
    keep (all | none);
    local-address address;
    local-as autonomous-system <private>;
    local-preference local-preference;
    log-updown;
    metric-out metric;
    multihop <ttl-value>;
    multipath {
      multiple-as;
    }
    no-aggregator-id;
    no-client-reflect;
    out-delay seconds;
    passive;
    peer-as autonomous-system;
    preference preference;
    remove-private;
    tcp-mss segment-size;
    traceoptions {
      file filename <files number> <size size> <world-readable | no-world-readable>;
      flag flag <flag-modifier> <disable>;
    }
    type type;
    neighbor address {
      ... peer-specific-options ...
    }
}
```

Hierarchy Level

[edit logical-systems *logical-system-name* protocols bgp],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp],
[edit protocols bgp],
[edit routing-instances *routing-instance-name* protocols bgp]

Release Information

Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description

Define a BGP peer group. BGP peer groups share a common type, peer autonomous system (AS) number, and cluster ID, if present. To configure multiple BGP groups, include multiple **group** statements.

By default, the group's options are identical to the global BGP options. To override the global options, include group-specific options within the **group** statement.

The **group** statement is one of the statements you must include in the configuration to run BGP on the routing device. See "Minimum BGP Configuration" on page 723.

Options *group-name*—Name of the BGP group.

The remaining statements are explained separately.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring BGP Groups and Peers on page 728

hold-time

Syntax
hold-time *seconds*;

Hierarchy Level
[edit logical-systems *logical-system-name* protocols bgp],
[edit logical-systems *logical-system-name* protocols bgp group *group-name*],
[edit logical-systems *logical-system-name* protocols bgp group *group-name* neighbor *address*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address*],
[edit protocols bgp],
[edit protocols bgp group *group-name*],
[edit protocols bgp group *group-name* neighbor *address*],
[edit routing-instances *routing-instance-name* protocols bgp],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name*],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address*]

Release Information
Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description
Specify the hold-time value to use when negotiating a connection with the peer. The hold-time value is advertised in open packets and indicates to the peer the length of time that it should consider the sender valid. If the peer does not receive a keepalive, update, or notification message within the specified hold time, the BGP connection to the peer is closed and routing devices through that peer become unavailable.

The hold time is three times the interval at which keepalive messages are sent.

Options
seconds—Hold time.
Range: 20 through 65,535 seconds
Default: 90 seconds

Required Privilege Level
routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring the Delay Before BGP Peers Mark the Routing Device as Down on page 741

idle-after-switch-over

Syntax idle-after-switch-over (forever | *seconds*);

Hierarchy Level [edit logical-systems *logical-system-name* protocols bgp],
[edit logical-systems *logical-system-name* protocols bgp group *group-name*],
[edit logical-systems *logical-system-name* protocols bgp group *group-name* neighbor *address*],
[edit protocols bgp],
[edit protocols bgp group *group-name*],
[edit protocols bgp group *group-name* neighbor *address*]

Release Information Statement introduced in JUNOS Release 9.5.
Statement introduced in JUNOS Release 9.5 for EX Series switches.

Description Configure the routing device not to automatically reestablish BGP peering sessions after a nonstop active routing (NSR) switchover. This feature is particularly useful if you are using dynamic routing policies because the dynamic database is not synchronized with the backup Routing Engine when NSR is enabled.

Options forever—Do not reestablish a BGP peering session after an NSR switchover until the clear bgp neighbor command is issued.

seconds—Do not reestablish a BGP peering session after an NSR switchover until after the specified period.
Range: 1 through 4,294,967,295 ($2^{32} - 1$)

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Preventing Automatic Reestablishment of BGP Peering Sessions After NSR Switchovers on page 786
- *JUNOS Policy Framework Configuration Guide*
- *JUNOS High Availability Configuration Guide*

import

Syntax import [*policy-names*];

Hierarchy Level
[edit logical-systems *logical-system-name* protocols bgp],
[edit logical-systems *logical-system-name* protocols bgp group *group-name*],
[edit logical-systems *logical-system-name* protocols bgp group *group-name* neighbor *address*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address*],
[edit protocols bgp],
[edit protocols bgp group *group-name*],
[edit protocols bgp group *group-name* neighbor *address*],
[edit routing-instances *routing-instance-name* protocols bgp],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name*],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address*]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Apply one or more routing policies to routes being imported into the JUNOS routing table from BGP.

Options *policy-names*—Name of one or more policies.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- export
- Applying Policies to BGP Routes on page 781
- *JUNOS Policy Framework Configuration Guide*

include-mp-next-hop

Syntax include-mp-next-hop;

Hierarchy Level [edit logical-systems *logical-system-name* protocols bgp],
[edit protocols bgp]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Enable multiprotocol updates to contain next-hop reachability information.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Including Next-Hop Reachability Information in Multiprotocol Updates on page 789

ipsec-sa

Syntax ipsec-sa *ipsec-sa*;

Hierarchy Level [edit logical-systems *logical-system-name* protocols bgp],
[edit logical-systems *logical-system-name* protocols bgp group *group-name*],
[edit logical-systems *logical-system-name* protocols bgp group *group-name* neighbor *address*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address*],
[edit protocols bgp],
[edit protocols bgp group *group-name*],
[edit protocols bgp group *group-name* neighbor *address*],
[edit routing-instances *routing-instance-name* protocols bgp],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name*],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address*]

Release Information Statement introduced before JUNOS Release 7.4.

Description Apply a security association to BGP peers. You can apply the security association globally for all BGP peers, to a group of peers, or to an individual peer.

Options *ipsec-sa*—Security association name.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Using IPsec to Protect BGP Traffic on page 745

iso-vpn

Syntax
```
iso-vpn {
    unicast {
        prefix-limit number;
        rib-group group-name;
    }
}
```

Hierarchy Level
[edit protocols bgp family],
[edit protocols bgp group *group-name* family],
[edit protocols bgp group *group-name* neighbor *address*family],
[edit routing-instances *routing-instance-name* protocols bgp family],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name* family],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address* family]

Release Information Statement introduced before JUNOS Release 7.4.

Description Enable BGP to carry ISO VPN NLRI messages between PE routes connecting a VPN.

NOTE: CLNS is supported for the J Series Services Router only.

The remaining statements are explained separately in this chapter.

Default Disabled.

Required Privilege Level
routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Enabling BGP to Carry CLNS Routes on page 774
- *JUNOS Software Interfaces and Routing Configuration Guide*

keep

Syntax keep (all | none);

Hierarchy Level
[edit logical-systems *logical-system-name* protocols bgp],
[edit logical-systems *logical-system-name* protocols bgp group *group-name*],
[edit logical-systems *logical-system-name* protocols bgp group *group-name* neighbor *address*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address*],
[edit protocols bgp],
[edit protocols bgp group *group-name*],
[edit protocols bgp group *group-name* neighbor *address*],
[edit routing-instances *routing-instance-name* protocols bgp],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name*],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address*]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Specify whether routes learned from a BGP peer are retained in the routing table even if they contain an AS number that was exported from the local AS.

Default If you do not include this statement, most routes are retained in the routing table.

Options all—Retain all routes.

none—Retain none of the routes. When keep none is configured for the BGP session and the inbound policy changes, the JUNOS Software forces readvertisement of the full set of routes advertised by the peer.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring How Often BGP Exchanges Routes with the Routing Table on page 784

labeled-unicast

Syntax

```
labeled-unicast {
    accepted-prefix-limit {
        maximum number;
        teardown <percentage> <idle-timeout (forever | minutes)>;
    }
    aggregate-label {
        community community-name;
    }
    explicit-null {
        connected-only;
    }
    prefix-limit {
        maximum number;
        teardown <percentage> <idle-timeout (forever | minutes)>;
    }
    resolve-vpn;
    rib inet.3;
    rib-group group-name;
}
```

Hierarchy Level

[edit logical-systems *logical-system-name* protocols bgp family (inet | inet6)],
[edit logical-systems *logical-system-name* protocols bgp group *group-name* family (inet | inet6)],
[edit logical-systems *logical-system-name* protocols bgp group *group-name* neighbor *address* family (inet | inet6)],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp family (inet | inet6)],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name* family (inet | inet6)],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address* family (inet | inet6)],
[edit protocols bgp family (inet | inet6)],
[edit protocols bgp group *group-name* family (inet | inet6)],
[edit protocols bgp group *group-name* neighbor *address* family (inet | inet6)],
[edit routing-instances *routing-instance-name* protocols bgp family (inet | inet6)],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name* family (inet | inet6)],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address* family (inet | inet6)]

Release Information

Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description

Configure the family type to be labeled-unicast.

The remaining statements are explained separately.

Required Privilege Level

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

Related Topics

- Enabling Multiprotocol BGP on page 767

local-address

Syntax local-address *address*;

Hierarchy Level
[edit logical-systems *logical-system-name* protocols bgp],
[edit logical-systems *logical-system-name* protocols bgp group *group-name*],
[edit logical-systems *logical-system-name* protocols bgp group *group-name* neighbor *address*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address*],
[edit protocols bgp],
[edit protocols bgp group *group-name*],
[edit protocols bgp group *group-name* neighbor *address*],
[edit routing-instances *routing-instance-name* protocols bgp],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name*],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address*]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Specify the address of the local end of a BGP session. This address is used to accept incoming connections to the peer and to establish connections to the remote peer. When none of the operational interfaces are configured with the specified local address, a session with a BGP peer is placed in the idle state.

Default If you do not configure a local address, BGP uses the routing device's source address selection rules to set the local address. For more information, see the *JUNOS Network Interfaces Configuration Guide*.

Options *address*—IPv6 or IPv4 address of the local end of the connection.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- router-id
- Assigning a BGP Identifier on page 725

local-as

Syntax local-as *autonomous-system* <loops *number*> <private | alias> <no-prepend-global-as>;

Hierarchy Level
[edit logical-systems *logical-system-name* protocols bgp],
[edit logical-systems *logical-system-name* protocols bgp group *group-name*],
[edit logical-systems *logical-system-name* protocols bgp group *group-name* neighbor *address*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address*],
[edit protocols bgp],
[edit protocols bgp group *group-name*],
[edit protocols bgp group *group-name* neighbor *address*],
[edit routing-instances *routing-instance-name* protocols bgp],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name*],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address*]

Release Information
Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.
alias option introduced in JUNOS Release 9.5.
no-prepend-global-as option introduced in JUNOS Release 9.6.

Description Set the local AS number.

In JUNOS Release 9.1 and later, the autonomous system (AS) numeric range in plain-number format is extended to provide BGP support for 4-byte AS numbers, as defined in RFC 4893, *BGP Support for Four-octet AS Number Space*.

In JUNOS Release 9.3 and later, you can also configure a 4-byte AS number using the AS-dot notation format of two integer values joined by a period: < *16-bit high-order value in decimal* > . < *16-bit low-order value in decimal* > . For example, the 4-byte AS number of 65546 in plain-number format is represented as 1.10 in the AS-dot notation format.

Options
alias—(Optional) Configure the local AS as an alias of the global AS number configured for the router at the [edit routing-options] hierarchy level. As a result, a BGP peer considers any local AS to which it is assigned as equivalent to the primary AS number configured for the routing device. When you use the alias option, only the AS (global or local) used to establish the BGP session is prepended in the AS path sent to the BGP neighbor.

autonomous-system—AS number.
Range: 1 through 4,294,967,295 (2^{32} – 1) in plain-number format
Range: 0.0 through 65535.65535 in AS-dot notation format

loops *number*—(Optional) Specify the maximum number of times that the local AS number can appear in an AS path received from a BGP peer. For *number*, include a value from 1 through 10.

no-prepend-global-as—(Optional) Specify to strip the global AS and to prepend only the local AS in AS paths sent to external peers.

private—(Optional) Configure to use the local AS only during the establishment of the BGP session with a BGP neighbor but to hide it in the AS path sent to external BGP peers. Only the global AS is included in the AS path sent to external peers.

NOTE: The private and alias options are mutually exclusive. You cannot configure both options with the same local-as statement.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- autonomous-system
- family
- Configuring a Local AS for EBGP Sessions on page 756

local-interface

Syntax local-interface *interface-name*;

Hierarchy Level [edit logical-systems *logical-system-name* protocols bgp group *group-name* neighbor *ipv6-link-local-address*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *ipv6-link-local-address*],
[edit protocols bgp group *group-name* neighbor *ipv6-link-local-address*],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *ipv6-link-local-address*]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Specify the interface name of the peer for IPv6 peering using link-local addresses. This peer is link-local in scope.

Options *interface-name*—Interface name of the EBGP IPv6 peer.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring EBGP Peering Using IPv6 Link-Local Addresses on page 786

local-preference

Syntax local-preference *local-preference*;

Hierarchy Level
[edit logical-systems *logical-system-name* protocols bgp],
[edit logical-systems *logical-system-name* protocols bgp group *group-name*],
[edit logical-systems *logical-system-name* protocols bgp group *group-name* neighbor *address*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address*],
[edit protocols bgp],
[edit protocols bgp group *group-name*],
[edit protocols bgp group *group-name* neighbor *address*],
[edit routing-instances *routing-instance-name* protocols bgp],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name*],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address*]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Modify the value of the LOCAL_PREF path attribute, which is a metric used by IBGP sessions to indicate the degree of preference for an external route. The route with the highest local preference value is preferred.

The LOCAL_PREF path attribute always is advertised to internal BGP peers and to neighboring confederations. It is never advertised to external BGP peers.

Default If you omit this statement, the LOCAL_PREF path attribute, if present, is not modified.

Options *local-preference*—Preference to assign to routes learned from BGP or from the group or peer.
Range: 0 through 4,294,967,295 ($2^{32} - 1$)
Default: If the LOCAL_PREF path attribute is present, do not modify its value. If a BGP route is received without a LOCAL_PREF attribute, the route is handled locally (it is stored in the routing table and advertised by BGP) as if it were received with a LOCAL_PREF value of 100. By default, non-BGP routes that are advertised by BGP are advertised with a LOCAL_PREF value of 100.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- preference
- Configuring the Local Preference Value for BGP Routes on page 752

log-updown

Syntax log-updown;

Hierarchy Level
[edit logical-systems *logical-system-name* protocols bgp],
[edit logical-systems *logical-system-name* protocols bgp group *group-name*],
[edit logical-systems *logical-system-name* protocols bgp group *group-name* neighbor *address*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address*],
[edit protocols bgp],
[edit protocols bgp group *group-name*],
[edit protocols bgp group *group-name* neighbor *address*],
[edit routing-instances *routing-instance-name* protocols bgp],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name*],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address*]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Log a message whenever a BGP peer makes a state transition. Messages are logged using the system logging mechanism located at the [edit system syslog] hierarchy level.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- traceoptions
- Configuring System Logging of BGP Peer State Transitions on page 788
- *JUNOS System Basics Configuration Guide*

metric-out

Syntax metric-out (*metric* | minimum-igp *offset* | igp (delay-med-update | *offset*);

Hierarchy Level
[edit logical-systems *logical-system-name* protocols bgp],
[edit logical-systems *logical-system-name* protocols bgp group *group-name*],
[edit logical-systems *logical-system-name* protocols bgp group *group-name* neighbor *address*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address*],
[edit protocols bgp],
[edit protocols bgp group *group-name*],
[edit protocols bgp group *group-name* neighbor *address*],
[edit routing-instances *routing-instance-name* protocols bgp],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name*],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address*]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.
delay-med-update option introduced in JUNOS Release 9.0.

Description Metric for all routes sent using the multiple exit discriminator (MED, or MULTI_EXIT_DISC) path attribute in update messages. This path attribute is used to discriminate among multiple exit points to a neighboring AS. If all other factors are equal, the exit point with the lowest metric is preferred.

You can specify a constant metric value by including the *metric* option. For configurations in which a BGP peer sends third-party next hops that require the local system to perform next-hop resolution—IBGP configurations, configurations within confederation peers, or EBGP configurations that include the multihop command—you can specify a variable metric by including the minimum-igp or igp option.

You can increase or decrease the variable metric calculated from the IGP metric (either from the igp or igp-minimum statement) by specifying a value for *offset*. The metric is increased by specifying a positive value for *offset*, and decreased by specifying a negative value for *offset*.

In JUNOS Release 9.0 and later, you can specify for a BGP group or peer not to advertise updates for the MED path attributes used to calculate IGP costs for BGP next hops unless the MED is lower. You can also configure an interval to delay when MED updates are sent by including the med-igp-update-interval *minutes* at the [edit routing-options] hierarchy level.

Options
delay-med-update—Specify for a BGP group or peer configured with the metric-out igp statement not to advertise MED updates when the value worsens, that is, unless the value is lower.

NOTE: You cannot configure delay-med–update statement at the global BGP level.

igp—Set the metric to the most recent metric value calculated in the IGP to get to the BGP next hop.

metric—Primary metric on all routes sent to peers.
Range: 0 through 4,294,967,295 ($2^{32} - 1$)
Default: No metric is sent.

minimum-igp—Set the metric to the minimum metric value calculated in the IGP to get to the BGP next hop. If a newly calculated metric is greater than the minimum metric value, the metric value remains unchanged. If a newly calculated metric is lower, the metric value is lowered to that value.

offset—(Optional) Increases or decreases the metric by this value.
Range: -2^{31} through $2^{31} - 1$
Default: None

Required Privilege Level
routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- med-igp-update-interval
- Configuring the MED in BGP Updates on page 746

mtu-discovery

Syntax mtu-discovery;

Hierarchy Level
[edit logical-systems *logical-system-name* protocols bgp],
[edit logical-systems *logical-system-name* protocols bgp group *group-name*],
[edit logical-systems *logical-system-name* protocols bgp group *group-name* neighbor *address*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address*],
[edit protocols bgp],
[edit protocols bgp group *group-name*],
[edit protocols bgp group *group-name* neighbor *address*],
[edit routing-instances *routing-instance-name* protocols bgp],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name*],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address*]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Configure TCP path maximum transmission unit (MTU) discovery. MTU discovery improves convergence times for IBGP sessions.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring MTU Discovery for BGP Sessions on page 741

multihop

Syntax

```
multihop {
    no-nexthop-change;
    ttl-value;
}
```

Hierarchy Level

[edit logical-systems *logical-system-name* protocols bgp],
[edit logical-systems *logical-system-name* protocols bgp group *group-name*],
[edit logical-systems *logical-system-name* protocols bgp group *group-name* neighbor *address*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address*],
[edit protocols bgp],
[edit protocols bgp group *group-name*],
[edit protocols bgp group *group-name* neighbor *address*],
[edit routing-instances *routing-instance-name* protocols bgp],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name*],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address*]

Release Information

Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description

Configure an EBGP multihop session.

External confederation peering is a special case that allows unconnected third-party next hops. You do not need to configure multihop sessions explicitly in this particular case; multihop behavior is implied.

If you have confederation external BGP peer-to-loopback addresses, you still need the multihop configuration.

Default

If you omit this statement, all EBGP peers are assumed to be directly connected (that is, you are establishing a nonmultihop, or "regular," BGP session), and the default time-to-live (TTL) value is 1.

Options

no-nexthop-change—Specify not to change the BGP next-hop value; for route advertisements, specify the no-nexthop-self option.

ttl-value—Configure the maximum TTL value for the TTL in the IP header of BGP packets.
Range: 1 through 255
Default: 64 (for multihop EBGP sessions, confederations, and IBGP sessions)

Required Privilege Level

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

Related Topics

- Configuring EBGP Multihop Sessions on page 750

multipath

Syntax

```
multipath {
    multiple-as;
}
```

Hierarchy Level

[edit logical-systems *logical-system-name* protocols bgp group *group-name*],
[edit logical-systems *logical-system-name* protocols bgp group *group-name* neighbor *address*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address*],
[edit protocols bgp group *group-name*],
[edit protocols bgp group *group-name* neighbor *address*],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name*],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address*]

Release Information

Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description

Allow load sharing among multiple EBGP paths and multiple IBGP paths.

Options

multiple-as—Disable the default check requiring that paths accepted by BGP multipath must have the same neighboring AS.

Required Privilege Level

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

Related Topics

- Selecting Multiple Equal-Cost Active Paths on page 755

neighbor

Syntax

```
neighbor address {
    accept-remote-nexthop;
    advertise-external <conditional>;
    advertise-inactive;
    (advertise-peer-as | no-advertise-peer-as);
    as-override;
    authentication-algorithm algorithm;
    authentication-key key;
    authentication-key-chain key-chain;
    cluster cluster-identifier;
    damping;
    description text-description;
    export [ policy-names ];
    family {
        (inet | inet6 | inet-mvpn | inet6-mpvn | inet-vpn | inet6-vpn | iso-vpn | l2-vpn) {
            (any | flow | multicast | unicast | signaling) {
                accepted-prefix-limit {
                    maximum number;
                    teardown <percentage> <idle-timeout (forever | minutes)>;
                }
                prefix-limit {
                    maximum number;
                    teardown <percentage> <idle-timeout (forever | minutes)>;
                }
                rib-group group-name;
            }
            flow {
                no-validate policy-name;
            }
            labeled-unicast {
                accepted-prefix-limit {
                    maximum number;
                    teardown <percentage> <idle-timeout (forever | minutes)>;
                }
                aggregate-label {
                    community community-name:
                }
                explicit-null {
                    connected-only;
                }
                prefix-limit {
                    maximum number;
                    teardown <percentage> <idle-timeout (forever | minutes)>;
                }
                resolve-vpn;
                rib inet.3;
                rib-group group-name;
            }
        }
        route-target {
            advertise-default;
```

```
            external-paths number;
            accepted-prefix-limit {
               maximum number;
               teardown <percentage> <idle-timeout (forever | minutes)>;
            }
            prefix-limit {
               maximum number;
               teardown <percentage> <idle-timeout (forever | minutes)>;
            }
         }
         signaling {
            prefix-limit {
               maximum number;
               teardown <percentage> <idle-timeout (forever | minutes)>;
            }
         }
      }
      graceful-restart {
         disable;
         restart-time seconds;
         stale-routes-time seconds;
      }
      hold-time seconds;
      import [ policy-names ];
      ipsec-sa ipsec-sa;
      keep (all | none);
      local-address address;
      local-as autonomous-system <private>;
      local-interface interface-name;
      local-preference preference;
      log-updown;
      metric-out (metric | minimum-igp <offset> | igp <offset>);
      mtu-discovery;
      multihop <ttl-value>;
      multipath {
         multiple-as;
      }
      no-aggregator-id;
      no-client-reflect;
      out-delay seconds;
      passive;
      peer-as autonomous-system;
      preference preference;
      tcp-mss segment-size;
      traceoptions {
         file filename <files number> <size size> <world-readable | no-world-readable>;
         flag flag <flag-modifier> <disable>;
      }
      vpn-apply-export;
   }
```

Hierarchy Level [edit logical-systems *logical-system-name* protocols bgp group *group-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name*],
[edit protocols bgp group *group-name*],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name*]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Explicitly configure a neighbor (peer). To configure multiple BGP peers, include multiple neighbor statements.

By default, the peer's options are identical to those of the group. You can override these options by including peer-specific option statements within the neighbor statement.

The neighbor statement is one of the statements you can include in the configuration to define a minimal BGP configuration on the routing device. (You can include an allow all statement in place of a neighbor statement.)

Options *address*—IPv6 or IPv4 address of a single peer.

The remaining statements are explained separately.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Minimum BGP Configuration on page 723
- Configuring BGP Groups and Peers on page 728

no-advertise-peer-as

See advertise-peer-as

no-aggregator-id

Syntax no-aggregator-id;

Hierarchy Level
[edit logical-systems *logical-system-name* protocols bgp],
[edit logical-systems *logical-system-name* protocols bgp group *group-name*],
[edit logical-systems *logical-system-name* protocols bgp group *group-name* neighbor *address*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address*],
[edit protocols bgp],
[edit protocols bgp group *group-name*],
[edit protocols bgp group *group-name* neighbor *address*],
[edit routing-instances *routing-instance-name* protocols bgp],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name*],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address*]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Set the router ID in the BGP aggregator path attribute to zero. (This is one of the path attributes included in BGP update messages.) Doing this prevents different routing devices within an AS from creating aggregate routes that contain different AS paths.

Default If you omit this statement, the router ID is included in the BGP aggregator path attribute.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Update Messages on page 717
- Controlling BGP Route Aggregation on page 750

no-client-reflect

Syntax no-client-reflect;

Hierarchy Level
[edit logical-systems *logical-system-name* protocols bgp],
[edit logical-systems *logical-system-name* protocols bgp group *group-name*],
[edit logical-systems *logical-system-name* protocols bgp group *group-name* neighbor *address*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address*],
[edit protocols bgp],
[edit protocols bgp group *group-name*],
[edit protocols bgp group *group-name* neighbor *address*],
[edit routing-instances *routing-instance-name* protocols bgp],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name*],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address*]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Disable intracluster route redistribution by the system acting as the route reflector. Include this statement when the client cluster is fully meshed to prevent the sending of redundant route advertisements.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- cluster
- Configuring BGP Route Reflection on page 761

no-validate

Syntax no-validate *policy-name*;

Hierarchy Level [edit protocols bgp group *group-name* family (inet | inet flow)],
[edit protocols bgp group *group-name* neighbor *address* family (inet | inet flow)],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name* family (inet | inet flow)],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address* family (inet | inet flow)]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Omits the flow route validation procedure after packets are accepted by a policy.

Options *policy-name*—Import policy to match NLRI messages.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Enabling BGP to Carry Flow-Specification Routes on page 773

out-delay

Syntax out-delay *seconds*;

Hierarchy Level
[edit logical-systems *logical-system-name* protocols bgp],
[edit logical-systems *logical-system-name* protocols bgp group *group-name*],
[edit logical-systems *logical-system-name* protocols bgp group *group-name* neighbor *address*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address*],
[edit protocols bgp],
[edit protocols bgp group *group-name*],
[edit protocols bgp group *group-name* neighbor *address*],
[edit routing-instances *routing-instance-name* protocols bgp],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name*],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address*]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Specify how long a route must be present in the JUNOS routing table before it is exported to BGP. Use this time delay to help bundle routing updates.

Default If you omit this statement, routes are exported to BGP immediately after they have been added to the routing table.

Options *seconds*—Output delay time.
Range: 0 through 65,535 seconds
Default: 0 seconds

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring How Often BGP Exchanges Routes with the Routing Table on page 784

outbound-route-filter

Syntax

```
outbound-route-filter {
    bgp-orf-cisco-mode;
    prefix-based {
        accept {
            (inet | inet6);
        }
    }
}
```

Hierarchy Level

[edit logical-systems *logical-system-name* protocols bgp],
[edit logical-systems *logical-system-name* protocols bgp group *group-name*],
[edit logical-systems *logical-system-name* protocols bgp group *group-name* neighbor *address*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address*],
[edit protocols bgp],
[edit protocols bgp group *group-name*],
[edit protocols bgp group *group-name* neighbor *address*],
[edit routing-instances *routing-instance-name* protocols bgp],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name*],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address*]

Release Information

Statement introduced in JUNOS Release 9.2.
Statement introduced in JUNOS Release 9.2 for EX Series switches.

Description

Configure a BGP peer to accept outbound route filters from a remote peer.

Options

accept—Specify that outbound route filters from a BGP peer be accepted.

inet—Specify that IPv4 prefix-based outbound route filters be accepted.

inet6—Specify that IPv6 prefix-based outbound route filters be accepted.

NOTE: You can specify that both IPv4 and IPv6 outbound route filters be accepted.

prefix-based—Specify that prefix-based filters be accepted.

The bgp-orf-cisco-mode statement is explained separately.

Required Privilege Level

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

Related Topics

- Applying Filters Provided by BGP Peers to Outbound Routes on page 780

passive

Syntax passive;

Hierarchy Level
[edit logical-systems *logical-system-name* protocols bgp],
[edit logical-systems *logical-system-name* protocols bgp group *group-name*],
[edit logical-systems *logical-system-name* protocols bgp group *group-name* neighbor *address*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address*],
[edit protocols bgp],
[edit protocols bgp group *group-name*],
[edit protocols bgp group *group-name* neighbor *address*],
[edit routing-instances *routing-instance-name* protocols bgp],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name*],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address*]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Do not send active open messages to the peer. Rather, wait for the peer to issue an open request.

Default If you omit this statement, all explicitly configured peers are active, and each peer periodically sends open requests until its peer responds.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Disabling Transmission of Open Requests to BGP Peers on page 746

path-selection

Syntax

```
path-selection {
   (always-compare-med | cisco-non-deterministic | external-router-id);
   med-plus-igp {
      igp-multiplier number;
      med-multiplier number;
   }
}
```

Hierarchy Level

[edit logical-systems *logical-system-name* protocols bgp],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp],
[edit protocols bgp],
[edit routing-instances *routing-instance-name* protocols bgp]

Release Information

Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.
med-plus-igp option introduced in JUNOS Release 8.1.

Description

Configure BGP path selection.

Default

If the path-selection statement is not included in the configuration, only the multiple exit discriminators (MEDs) of routes that have the same peer ASs are compared.

Options

cisco-non-deterministic—Configure routing table path selection so that it is performed using the same nondeterministic behavior as the Cisco IOS software. The active path is always first. All inactive, but eligible, paths follow the active path and are maintained in the order in which they were received, with the most recent path first. Ineligible paths remain at the end of the list.

always-compare-med—Always compare MEDs whether or not the peer ASs of the compared routes are the same.

NOTE: We recommend that you configure the always-compare-med option.

external-router-id—Compare the router ID between external BGP paths to determine the active path.

med-plus-igp—Add the IGP cost to the next-hop destination to the MED before comparing MED values for path selection.

igp-multiplier *number*—The multiplier value for the IGP cost to a next-hop address.
Range: 1 through 1000
Default: None

med-multiplier *number*—The multiplier value for the MED calculation.
Range: 1 through 1000
Default: None

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring Routing Table Path Selection for BGP on page 754

peer-as

Syntax peer-as *autonomous-system*;

Hierarchy Level
[edit logical-systems *logical-system-name* protocols bgp],
[edit logical-systems *logical-system-name* protocols bgp group *group-name*],
[edit logical-systems *logical-system-name* protocols bgp group *group-name* neighbor *address*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address*],
[edit protocols bgp],
[edit protocols bgp group *group-name*],
[edit protocols bgp group *group-name* neighbor *address*],
[edit routing-instances *routing-instance-name* protocols bgp],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name*],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address*]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Specify the neighbor (peer) AS number.

The autonomous system (AS) numeric range in plain-number format has been extended in JUNOS Release 9.1 to provide BGP support for 4-byte AS numbers, as defined in RFC 4893, *BGP Support for Four-octet AS Number Space*.

In JUNOS Release 9.2 and later, you can also configure a 4-byte AS number using the AS-dot notation format of two integer values joined by a period: *<16-bit high-order value in decimal>.<16-bit low-order value in decimal>*. For example, the 4-byte AS number of 65,546 in plain-number format is represented as 1.10 in the AS-dot notation format.

Options *autonomous-system*—AS number.
Range: 1 through 4,294,967,295 ($2^{32} - 1$) in plain-number format
Range: 0.0 through 65535.65535 in AS-dot notation format

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring BGP Groups and Peers on page 728
- Specifying the Peer's AS Number on page 731

preference

Syntax preference *preference*;

Hierarchy Level
[edit logical-systems *logical-system-name* protocols bgp],
[edit logical-systems *logical-system-name* protocols bgp group *group-name*],
[edit logical-systems *logical-system-name* protocols bgp group *group-name* neighbor *address*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address*],
[edit protocols bgp],
[edit protocols bgp group *group-name*],
[edit protocols bgp group *group-name* neighbor *address*],
[edit routing-instances *routing-instance-name* protocols bgp],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name*],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address*]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Specify the preference for routes learned from BGP.

At the BGP global level, the preference statement sets the preference for routes learned from BGP. You can override this preference in a BGP group or peer preference statement.

At the group or peer level, the preference statement sets the preference for routes learned from the group or peer. Use this statement to override the preference set in the BGP global preference statement when you want to favor routes from one group or peer over those of another.

Options *preference*—Preference to assign to routes learned from BGP or from the group or peer.
Range: 0 through 4,294,967,295 ($2^{32} - 1$)
Default: 170 for the primary preference

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- local-preference
- Configuring the Default Preference Value for BGP Routes on page 752

prefix-limit

Syntax

```
prefix-limit {
    maximum number;
    teardown <percentage> <idle-timeout (forever | minutes)>;
}
```

Hierarchy Level

[edit logical-systems *logical-system-name* protocols bgp family (inet | inet6) (any | flow | labeled-unicast | multicast | unicast)],
[edit logical-systems *logical-system-name* protocols bgp group *group-name* family (inet | inet6) (any | flow | labeled-unicast | multicast | unicast)],
[edit logical-systems *logical-system-name* protocols bgp group *group-name* neighbor *address* family (inet | inet6) (any | flow | labeled-unicast | multicast | unicast)],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp family (inet | inet6) (any | flow | labeled-unicast | multicast | unicast)],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name* family (inet | inet6) (any | flow | labeled-unicast | multicast | unicast)],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address* family (inet | inet6) (any | flow | labeled-unicast | multicast | unicast)],
[edit protocols bgp family (inet | inet6) (any | flow | labeled-unicast | multicast | unicast)],
[edit protocols bgp group *group-name* family (inet | inet6) (any | labeled-unicast | multicast | unicast)],
[edit protocols bgp group *group-name* neighbor *address* family (inet | inet6) (any | flow | labeled-unicast | multicast | unicast)],
[edit routing-instances *routing-instance-name* protocols bgp family (inet | inet6) (any | flow | labeled-unicast | multicast | unicast)],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name* family (inet | inet6) (any | flow | labeled-unicast | multicast | unicast)],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address* family (inet | inet6) (any | flow | labeled-unicast | multicast | unicast)]

Release Information

Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description

Limit the number of prefixes received on a BGP peering session and a rate-limit logging when injected prefixes exceed a set limit.

Options

maximum *number*—When you set the maximum number of prefixes, a message is logged when that number is exceeded.
Range: 1 through 4,294,967,295 ($2^{32} - 1$)

teardown <*percentage*>—If you include the teardown statement, the session is torn down when the maximum number of prefixes is reached. If you specify a percentage, messages are logged when the number of prefixes exceeds that percentage. After the session is torn down, it is reestablished in a short time unless you include the idle-timeout statement. Then the session can be kept down for a specified amount of time, or forever. If you specify forever, the session is reestablished only after you issue a clear bgp neighbor command.
Range: 1 through 100

idle-timeout (forever | *timeout-in-minutes*)—(Optional) If you include the idle-timeout statement, the session is torn down for a specified amount of time, or forever. If you specify a period of time, the session is allowed to reestablish after this timeout period. If you specify forever, the session is reestablished only after you intervene with a clear bgp neighbor command.
Range: 1 through 2400

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- accepted-prefix-limit
- Limiting the Number of Prefixes Received on a BGP Peering Session on page 771

remove-private

Syntax remove-private;

Hierarchy Level
[edit logical-systems *logical-system-name* protocols bgp],
[edit logical-systems *logical-system-name* protocols bgp group *group-name*],
[edit logical-systems *logical-system-name* protocols bgp group *group-name* neighbor *address*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address*],
[edit protocols bgp],
[edit protocols bgp group *group-name*],
[edit protocols bgp group *group-name* neighbor *address*],
[edit routing-instances *routing-instance-name* protocols bgp],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name*],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address*]

Release Information
Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description When advertising AS paths to remote systems, have the local system strip private AS numbers from the AS path. The numbers are stripped from the AS path starting at the left end of the AS path (the end where AS paths have been most recently added). The routing device stops searching for private ASs when it finds the first nonprivate AS. This operation takes place after any confederation member ASs have already been removed from the AS path, if applicable.

The JUNOS Software recognizes the set of AS numbers that is considered private, a range that is defined in the Internet Assigned Numbers Authority (IANA) assigned numbers document.

The set of reserved AS numbers is in the range from 64,512 through 65,535.

Required Privilege Level
routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Removing Private AS Numbers from AS Paths on page 761

resolve-vpn

Syntax resolve-vpn;

Hierarchy Level
[edit logical-systems *logical-system-name* protocols bgp family inet labeled-unicast],
[edit logical-systems *logical-system-name* protocols bgp group *group-name* family inet labeled-unicast],
[edit logical-systems *logical-system-name* protocols bgp group *group-name* neighbor *address* family inet labeled-unicast],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp family inet labeled-unicast],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name* family inet labeled-unicast],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address* family inet labeled-unicast],
[edit protocols bgp family inet labeled-unicast],
[edit protocols bgp group *group-name* family inet labeled-unicast],
[edit protocols bgp group *group-name* neighbor *address* family inet labeled-unicast],
[edit routing-instances *routing-instance-name* protocols bgp family inet labeled-unicast],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name* family inet labeled-unicast],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address* family inet labeled-unicast]

Release Information Statement introduced before JUNOS Release 7.4.

Description Allow labeled routes to be placed in the inet.3 routing table for route resolution. These routes are then resolved for PE router connections where the remote PE is located across another AS. For a PE router to install a route in the VRF, the next hop must resolve to a route stored within the inet.3 table.

Required Privilege Level
routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Enabling Multiprotocol BGP on page 767

rib

Syntax rib inet.3;

Hierarchy Level
[edit logical-systems *logical-system-name* protocols bgp family inet labeled-unicast],
[edit logical-systems *logical-system-name* protocols bgp group *group-name* family inet labeled-unicast],
[edit logical-systems *logical-system-name* protocols bgp group *group-name* neighbor *address* family inet labeled-unicast],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp family inet labeled-unicast],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name* family inet labeled-unicast],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address* family inet labeled-unicast],
[edit protocols bgp family inet labeled-unicast],
[edit protocols bgp group *group-name* family inet labeled-unicast],
[edit protocols bgp group *group-name* neighbor *address* family inet labeled-unicast],
[edit routing-instances *routing-instance-name* protocols bgp family inet labeled-unicast],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name* family inet labeled-unicast],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address* inet labeled-unicast]

Release Information Statement introduced before JUNOS Release 7.4.

Description You can allow both labeled and unlabeled routes to be exchanged in a single session. The labeled routes are placed in the inet.3 routing table, and both labeled and unlabeled unicast routes can be sent or received by the router.

Options inet.3—Name of the routing table.

Required Privilege Level
routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Enabling Multiprotocol BGP on page 767

rib-group

Syntax rib-group *group-name*;

Hierarchy Level
[edit logical-systems *logical-system-name* protocols bgp family inet (any | labeled-unicast | unicast | multicast)],
[edit logical-systems *logical-system-name* protocols bgp group *group-name* family inet (any | labeled-unicast | unicast | multicast)],
[edit logical-systems *logical-system-name* protocols bgp group *group-name* neighbor *address* family inet (any | labeled-unicast | unicast | multicast)],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp family inet (any | labeled-unicast | unicast | multicast)],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name* family inet (any | labeled-unicast | unicast | multicast)],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address* family inet (any | labeled-unicast | unicast | multicast)],
[edit protocols bgp family inet (any | labeled-unicast | unicast | multicast)],
[edit protocols bgp group *group-name* family inet (any | labeled-unicast | unicast | multicast)],
[edit protocols bgp group *group-name* neighbor *address* family inet (any | labeled-unicast | unicast | multicast)],
[edit routing-instances *routing-instance-name* protocols bgp family inet (any | labeled-unicast | unicast | multicast)],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name* family inet (any | labeled-unicast | unicast | multicast)],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address* family inet (any | labeled-unicast | unicast | multicast)]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Add unicast prefixes to unicast and multicast tables.

Options *group-name*—Name of the routing table group. The name must start with a letter and can include letters, numbers, and hyphens. You generally specify only one routing table group.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- interface-routes
- rib-group
- Creating Routing Table Groups on page 114
- Configuring How Interface Routes Are Imported into Routing Tables on page 116
- Configuring BGP Routing Table Groups on page 772

route-target

Syntax

```
route-target {
    accepted-prefix-limit {
        maximum number;
        teardown <percentage> <idle-timeout (forever | time-in-minutes)>;
    }
    advertise-default;
    external-paths number;
    prefix-limit {
        maximum number;
        teardown <percentage> <idle-timeout (forever | time-in-minutes)>;
    }
}
```

Hierarchy Level

[edit logical-systems *logical-system-name* protocols bgp family],
[edit logical-systems *logical-system-name* protocols bgp group *group-name* family],
[edit logical-systems *logical-system-name* protocols bgp group *group-name* neighbor *address* family],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name* family],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address* family],
[edit protocols bgp family],
[edit protocols bgp group *group-name* family],
[edit protocols bgp group *group-name* neighbor *address* family],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name* family],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address* family]

Release Information

Statement introduced before JUNOS Release 7.4.

Description

Limit the number of prefixes advertised on BGP peerings specifically to the peers that need the updates.

Options

advertise-default—Advertise default routes and suppress more specific routes.

external-paths *number*—Number of external paths accepted for route filtering.
Range: 1 through 16 paths
Default: 1 path

The remaining statements are explained separately.

Required Privilege Level

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

Related Topics

- Enabling BGP Route Target Filtering on page 779

tcp-mss

Syntax tcp-mss *segment-size*;

Hierarchy Level [edit logical-systems *logical-system-name* protocols bgp],
[edit logical-systems *logical-system-name* protocols bgp group *group-name*],
[edit logical-systems *logical-system-name* protocols bgp group *group-name* neighbor *neighbor-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *neighbor-name*],
[edit protocols bgp],
[edit protocol bgp group *group-name*],
[edit protocols bgp group *group-name* neighbor *neighbor-name*],
[edit routing-instances *routing-instance-name* protocols bgp],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name*],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *neighbor-name*]

Release Information Statement introduced in JUNOS Release 8.1.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Configure the maximum segment size (MSS) for the TCP connection for BGP neighbors.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Limiting TCP Segment Size for BGP on page 798

traceoptions

Syntax

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

Hierarchy Level

[edit logical-systems *logical-system-name* protocols bgp],
[edit logical-systems *logical-system-name* protocols bgp group *group-name*],
[edit logical-systems *logical-system-name* protocols bgp group *group-name* neighbor *address*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address*],
[edit protocols bgp],
[edit protocols bgp group *group-name*],
[edit protocols bgp group *group-name* neighbor *address*],
[edit routing-instances *routing-instance-name* protocols bgp],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name*],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name* neighbor *address*]

Release Information

Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.
4byte-as statement introduced in JUNOS Release 9.2.
4byte-as statement introduced in JUNOS Release 9.2 for EX Series switches.

Description

Configure BGP protocol-level tracing options. To specify more than one tracing operation, include multiple flag statements.

Default

The default BGP protocol-level tracing options are inherited from the routing protocols traceoptions statement included at the [edit routing-options] hierarchy level. The default group-level trace options are inherited from the BGP protocol-level traceoptions statement. The default peer-level trace options are inherited from the group-level traceoptions statement.

Options

disable—(Optional) Disable the tracing operation. You can use this option is to disable a single operation when you have defined a broad group of tracing operations, such as all.

file *name*—Name of the file to receive the output of the tracing operation. Enclose the name within quotation marks. All files are placed in the directory /var/log. We recommend that you place BGP tracing output in the file bgp-log.

files *number*—(Optional) Maximum number of trace files. When a trace file named *trace-file* reaches its maximum size, it is renamed *trace-file*.0, then *trace-file*.1, and so on, until the maximum number of trace files is reached. Then, the oldest trace file is overwritten.

If you specify a maximum number of files, you must also specify a maximum file size with the size option.
Range: 2 through 1000 files
Default: 10 files

flag—Tracing operation to perform. To specify more than one tracing operation, include multiple flag statements.

BGP Tracing Flags

- 4byte-as—4-byte AS events
- bfd—BFD protocol events
- damping—Damping operations
- graceful-restart—Graceful restart events
- keepalive—BGP keepalive messages. If you enable the the BGP update flag only, received keepalive messages do not generate a trace message.
- nsr-synchronization—Nonstop routing synchronization events
- open—Open packets. These packets are sent between peers when they are establishing a connection.
- packets—All BGP protocol packets
- refresh—BGP refresh packets
- update—Update packets. These packets provide routing updates to BGP systems. If you enable only this flag, received keepalive messages do not generate a trace message. Use the keepalive flag to generate a trace message for keepalive messages.

Global Tracing Flags

- all—All tracing operations
- general—A combination of the normal and route trace operations
- normal—All normal operations

  Default: If you do not specify this option, only unusual or abnormal operations are traced.
- policy—Policy operations and actions
- route—Routing table changes
- state—State transitions
- task—Routing protocol task processing
- timer—Routing protocol timer processing

flag-modifier—(Optional) Modifier for the tracing flag. You can specify one or more of these modifiers:

- detail—Provide detailed trace information
- filter—Filter trace information. Applies only to route and damping tracing flags.
- receive—Packets being received.
- send—Packets being transmitted.

no-world-readable—(Optional) Prevent any user from reading the log file.

size *size*—(Optional) Maximum size of each trace file, in kilobytes (KB), megabytes (MB), or gigabytes (GB). When a trace file named *trace-file* reaches this size, it is renamed *trace-file*.0. When the *trace-file* again reaches its maximum size, *trace-file*.0 is renamed *trace-file*.1 and *trace-file* is renamed *trace-file*.0. This renaming scheme continues until the maximum number of trace files is reached. Then, the oldest trace file is overwritten.
If you specify a maximum file size, you also must specify a maximum number of trace files with the files option.
Syntax: *x*k to specify KB, *x*m to specify MB, or *x*g to specify GB
Range: 10 KB through the maximum file size supported on your system
Default: 128 KB

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

Required Privilege Level routing and trace—To view this statement in the configuration.
routing-control and trace-control—To add this statement to the configuration.

Related Topics
- log-updown statement
- Tracing BGP Protocol Traffic on page 799
- Configuring OSPF Refresh and Flooding Reduction in Stable Topologies on page 468

type

Syntax type *type*;

Hierarchy Level [edit logical-systems *logical-system-name* protocols bgp group *group-name*],
[edit logical-systems *logical-system-name* routing-instances *routing-instance-name* protocols bgp group *group-name*],
[edit protocols bgp group *group-name*],
[edit routing-instances *routing-instance-name* protocols bgp group *group-name*]

Release Information Statement introduced before JUNOS Release 7.4.
Statement introduced in JUNOS Release 9.0 for EX Series switches.

Description Specify the type of BGP peer group.

Options *type*—Type of group:

- external—External group
- internal—Internal group

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Configuring BGP Groups and Peers on page 728

vpn-apply-export

Syntax vpn-apply-export;

Hierarchy Level [edit logical-systems *logical-system-name* protocols bgp],
[edit logical-systems *logical-system-name* protocols bgp group *group-name*],
[edit logical-systems *logical-system-name* protocols bgp group *group-name* neighbor *address*],
[edit protocols bgp],
[edit protocols bgp group *group-name*],
[edit protocols bgp group *group-name* neighbor *address*]

Release Information Statement introduced before JUNOS Release 7.4.

Description Apply a BGP export policy in addition to a VPN routing and forwarding (VRF) export policy to routes.

Default The default action is to accept.

Required Privilege Level routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

Related Topics
- Applying BGP Export Policy to VRF Routes on page 789

Part 7
Indexes

- Index on page 881
- Index of Statements and Commands on page 901

Index

Symbols

#, comments in configuration statements.....xlii
( ), in syntax descriptions.....xlii
< >, in syntax descriptions.....xli
[ ], in configuration statements.....xlii
{ }, in configuration statements.....xlii
| (pipe), in syntax descriptions.....xlii

A

accept
  firewall filters
    action.....108
accept-remote-nexthop statement.....804
  usage guidelines.....750
accepted-prefix-limit statement.....805
  usage guidelines.....771
access-profile statement
  routing instances.....273
action modifiers, firewall filters.....108
active aggregate routes.....87
active routes.....6, 7
active statement
  aggregate routes.....142
    usage guidelines.....94
  generated routes.....142
    usage guidelines.....102
  static routes.....142
    usage guidelines.....72
address statement.....665
  usage guidelines.....662
advertise statement.....666
  usage guidelines.....662
advertise-external statement.....807
  usage guidelines.....783
advertise-inactive statement.....808
  usage guidelines.....783
advertise-peer-as statement.....809
  usage guidelines.....785
advertise-unnumbered-interfaces statement
  OSPF
    usage guidelines.....481
aggregate routes.....87, 143
  preferences.....8
aggregate statement.....143
  usage guidelines.....87
aggregate-label statement.....810
  usage guidelines.....743
aggregator path attribute, BGP *See* BGP, aggregator path attribute
aggregator statement.....146
alert (system logging severity level).....192
all (tracing flag).....220
allow statement.....811
  usage guidelines.....730
alternate preferences.....6
always-compare-med option.....754
any-sender statement
  RIP.....605
  usage guidelines.....600
area statement.....500
  usage guidelines.....450
area-range statement.....501
  usage guidelines.....463
AS external link advertisements.....438
as-override statement.....812
as-path (tracing flag).....875
as-path statement.....146
  aggregate routes
    usage guidelines.....92
  generated routes
    usage guidelines.....100
  static routes
    usage guidelines.....69
ASs.....45
  configuring.....112, 149, 724
  paths.....715
    aggregate routes.....92, 146
    generated routes.....146, 159
    operations, tracing.....875
    static routes.....69, 146
  private, removing.....761, 869
auth (tracing flag).....627
authentication
  algorithm
    BGP.....743
  BGP.....717
  keychains
    BGP.....743

MD5
BGP....743
OSPF....458
OSPFv3....462
simple
RIP....587
authentication configuration
BFD....81, 333, 473, 592, 794
authentication-algorithm statement
BGP....813
usage guidelines....743
authentication-key statement
BGP....814
usage guidelines....743
IS-IS....374
usage guidelines....325
RIP....606
usage guidelines....587
authentication-key-chain statement....815
BGP
usage guidelines....743
usage guidelines....743
authentication-type statement
IS-IS....375
usage guidelines....325
RIP....607
usage guidelines....587
auto-export
routing instance....148
auto-export statement....148
usage guidelines....261
autonomous statement....685
usage guidelines....682
autonomous systems *See* ASs
autonomous-system statement....149
usage guidelines....112, 724

B

backup-spf-options statement
OSPF....504
bandwidth-based metrics
OSPF....465
bandwidth-based-metrics statement....505
usage guidelines....465
BFD
authentication
configuration....81, 333, 473, 592, 794
protocol....74, 328, 469, 588, 789
traceoptions statement
usage guidelines....78
bfd statement....151
bfd-liveness-detection statement
BGP....816
usage guidelines....789
IS-IS....376
usage guidelines....328
OSPF....507
usage guidelines....469
RIP....608
usage guidelines....588
static routes....154
usage guidelines....65, 74
BGP
aggregator path attribute....750, 858
AS numbers, peers....865
ASs *See* ASs
authentication....717, 743, 814
authentication algorithm....813
authentication keychain....815
autonomous system override....812
best external route
advertising....783
BFD....789, 816
CLNS....774
communities
aggregate routes....91
generated routes....99
static routes....68
confederations....161, 724, 737
configuration statements....722, 723, 803
configuring EBGP groups....729
description....788
EBGP IPv6 peering....786
enabling on router....724, 737, 820
external (EBGP)....715
graceful restart....833
groups....728, 834
hold time....717, 741, 837
identifier....717, 725
idle-after-switch-over statement....838
internal (IBGP)....715
IP address....717
IPsec....745, 840
keepalive messages....718, 875
local address....746, 844
local AS....756
local interface....847
messages....716
MP-BGP....767, 829
MTU discovery....852
multihop sessions....853
multipath configuration....755
multiprotocol reachability....789
Multitopology Routing
configuring....299
neighbors BGP, peers *See* BGP, peers

NLRI....717
IPv4 VPN....767
IPv6 VPN....767
open messages....717, 746, 863
outbound route filters
interoperability....821
overview....714
packets, tracing....875
passive mode....746
path attributes....715, 717
peers....715, 728, 855
policy, routing....828, 839
precedence....782
preferences....8, 752, 866
prefix-limit
accepted....771
received....771
resolve routes to other tables....772
route reflection....761, 823, 859
route target filtering....779, 873
router identifier....113, 211
routes....715
routing instances
configure....240
routing tables
delays in exchanging routes....784, 861
nonactive routes....783, 808
retaining routes....842
set local AS number....845
standards supported....718
system log messages....788
TCP....714
TCP port block....789
tracing operations....875
4-Byte AS events....799
BFD protocol events....799
damping operations....799
description....799
graceful restart....799
keepalive messages....799
NSR synchronization....799
open PDUs....799
policy processing....799
protocol task processing....799
protocol timer processing....799
refresh PDUs....799
route information....799
state transitions....799
update PDUs....799
type, group....878
update messages....717
version supported....714
VRF export policy....789
BGP Monitoring Protocol....822
configuring....798
bgp statement....820
usage guidelines....723
bgp-orf-cisco-mode statement....821
bmp statement....822
usage guidelines....798
BOOTP
accepting packets....123
Border Gateway Protocol *See* BGP
braces, in configuration statements....xlii
brackets
angle, in syntax descriptions....xli
square, in configuration statements....xlii
brief statement....158
aggregate routes
usage guidelines....93
generated routes
usage guidelines....101
broadcast mode, router discovery....662
broadcast statement....666
usage guidelines....662

C

check-zero statement....610
usage guidelines....595
checksum statement....378
usage guidelines....336
cisco-non-deterministic option....754, 864
class of service-based forwarding, configure....259
CLNS....841
BGP....774
static routes....64
clns-routing statement
IS-IS....378
usage guidelines....356
cluster statement....823
usage guidelines....761
color statement
aggregate routes....196
usage guidelines....90
generated routes....196
usage guidelines....99
static routes....196
usage guidelines....68
comments, in configuration statements....xlii
communities
aggregate routes....91, 159
generated routes....99
static routes....68, 159
community statement
aggregate routes....159
usage guidelines....91
generated routes....159
usage guidelines....99

Multitopology Routing....304
usage guidelines....299
static routes....159
usage guidelines....68
complete sequence number PDUs, IS-IS *See* IS-IS, complete sequence number PDUs
confederation statement....161
usage guidelines....113, 724
confederations....161, 724, 737
config-internal (tracing flag)....220
configuration mode, CLI
statement hierarchy....15
contributing routes
aggregate routes....87
generated routes....96
conventions
text and syntax....xli
count (firewall filter action)....108
credibility-protocol-preference
OSPF
usage guidelines....481
traffic engineering
IS-IS....429
critical (system logging severity level)....192
cryptographic-address statement....703
usage guidelines....700
csn (tracing flag)....427
csnp-interval statement....379
usage guidelines....336
curly braces, in configuration statements....xlii
current-hop-limit statement....686
usage guidelines....679
customer support....xlii
contacting JTAC....xlii

D

damping....824, 875
damping (tracing flag)....875
damping statement....824
usage guidelines....767
database description packets....437
dead-interval statement....511
usage guidelines....467
debug (system logging severity level)....192
default-lifetime statement....686
usage guidelines....679
default-lsa statement....512
usage guidelines....451
default-metric statement....513
usage guidelines....450
defaults statement
aggregate statement....143
usage guidelines....87
generate statement....171
usage guidelines....96
static statement....215
usage guidelines....54
delay statement
IS-IS....424
OSPF....565
demand-circuit statement....514
OSPF
usage guidelines....456
usage guidelines....488
description statement....274, 825
usage guidelines....788
designated router
IS-IS....345
OSPF....439, 462
destination-networks statement....161
usage guidelines....125
destination-port (firewall filter match condition)....106
detection-time statement
BFD....154
BGP....816
IS-IS....376
DHCP
accepting....123
disable statement
BGP....826
usage guidelines....725
graceful restart
usage guidelines....124
IS-IS....381
graceful restart....385
LDP synchronization....382
usage guidelines....342, 353
OSPF....517
LDP synchronization....516
usage guidelines....444
router discovery....667
routing options
usage guidelines....118
disabling multicast on an interface....118
discard (firewall filter action)....108
discard statement
aggregate routes....163
generated routes....163
discard statement, in static statement
usage guidelines....57
documentation
comments on....xlii
domain-id statement....518
usage guidelines....266
domain-vpn-tag statement....518
usage guidelines....266

dscp (firewall filter match condition).....106
Dynamic Host Configuration Protocol DHCP *See* DHCP
dynamic tunnels.....164
source.....213
dynamic-tunnels statement.....164
usage guidelines.....125

E

EBGP *See* BGP
EBGP IPv6 peering, BGP.....786
emergency (system logging severity level).....192
enable statement
routing options.....180
usage guidelines.....118
enabling multicast on an interface.....118
equal-cost paths.....6, 10
error (system logging severity level).....192
error (tracing flag)
IS-IS.....427
neighbor discovery.....694, 697
OSPF.....570
RIP.....627
RIPng.....655
Ethernet interfaces, unnumbered
as next-hop interface for static routes.....58
configuration example.....61
except (firewall filter match condition).....106
expiration (tracing flag).....655
neighbor discovery.....694, 697
explicit-null statement.....827
usage guidelines.....742
export statement
BGP.....828
usage guidelines.....783
forwarding table.....164
usage guidelines.....116, 120
IS-IS.....382
usage guidelines.....360
OSPF.....519
usage guidelines.....484
RIP.....611
usage guidelines.....599
RIPng.....643
usage guidelines.....638
export-rib statement.....165
usage guidelines.....114
external-preference statement
IS-IS.....383
usage guidelines.....347
OSPF.....520
usage guidelines.....465
external-router-id option.....754

F

family statement
BGP.....829
usage guidelines.....767
IS-IS.....384
usage guidelines.....351
fate-sharing statement.....166
usage guidelines.....45
FBF, configuring.....258
filter statement.....167
usage guidelines.....109
filter-based forwarding
configuring.....258
Multitopology Routing.....300
flash (tracing flag).....220
flood-reduction statement.....521
flooding (tracing flag).....570
flow routes.....168
BGP.....773
flow statement.....168, 829
usage guidelines.....105, 773
font conventions.....xli
forwarding table
aggregate routes.....93, 94, 158
generated routes.....101, 158
overview.....5
policy, routing.....164
static routes.....54, 71, 72, 94, 102, 142, 177, 203
synchronizing.....5
forwarding-cache statement.....169
usage guidelines.....119
forwarding-class (firewall filter action).....108
forwarding-table statement.....169
usage guidelines.....120
fragment-offset (firewall filter match condition).....106
full statement.....158
aggregate routes
usage guidelines.....93
generated routes
usage guidelines.....101

G

general (tracing flag).....220
neighbor discovery.....694, 697
RIPng.....655
generate statement.....171
usage guidelines.....96, 97
generated routes.....45, 171
preferences.....8
graceful restart.....124
graceful-restart (tracing flag)
IS-IS.....427
OSPF.....570

graceful-restart statement
BGP....833
usage guidelines....742
IS-IS....385
usage guidelines....350
OSPF....522
usage guidelines....477
RIP....612
usage guidelines....600
RIPng....644
usage guidelines....639
usage guidelines....124
group statement
BGP....834
usage guidelines....730
RIP....613
usage guidelines....598
RIPng....645
usage guidelines....637
usage guidelines....732

H

hello (tracing flag)
IS-IS....427
hello packets
IS-IS....317
OSPF....437
hello-authentication-key statement....386
IS-IS
usage guidelines....343
hello-authentication-type statement....387
usage guidelines....343
hello-interval statement
IS-IS....388
usage guidelines....343
OSPF....523
usage guidelines....466
hello-padding statement....389
usage guidelines....355
helper-disable statement
IS-IS....385
usage guidelines....350
hold-time statement
BGP....837
usage guidelines....741
IS-IS....390
LDP synchronization....391
usage guidelines....344
OSPF
LDP synchronization....524
holddown (tracing flag)....627, 655
neighbor discovery....694, 697
holddown statement
IS-IS....424
OSPF....565
RIP....615
usage guidelines....597
RIPng....646
usage guidelines....637
holddown-interval statement
BFD
static routes....154
BFD (static routes
usage guidelines....74

I

IBGP *See* BGP
ICMP router discovery....661
icmp-code (firewall filter match condition)....106
icmp-type (firewall filter match condition)....106
icons defined, notice....xl
identifiers
BGP *See* BGP, identifier
router *See* router identifier
idle-after-switch-over statement....838
ignore statement....666, 667
usage guidelines....662
ignore-attached-bit statement....391
usage guidelines....321
ignore-lsp-metrics statement
IS-IS....392
usage guidelines....352
OSPF....524
usage guidelines....481
import statement
BGP....839
usage guidelines....782
OSPF....525
usage guidelines....484
RIP....616
usage guidelines....596
RIPng....647
usage guidelines....636
route resolution....173
usage guidelines....127
import-policy statement....173
usage guidelines....114
import-rib statement....174
usage guidelines....114
include-mp-next-hop statement....840
usage guidelines....789
independent-domain statement....175
indirect next hop....128, 176
indirect-next-hop statement....176
usage guidelines....128
ineligible statement....667, 673
inet.0 routing table....4
inet.1 routing table....4
inet.2 routing table....4, 52
inet.3 routing table....4

inet6.0 routing table....4, 52
info (system logging severity level)....192
input statement....176
usage guidelines....109
install statement....177
usage guidelines....71
instance type, configuring....253
instance-export statement....178
usage guidelines....261
instance-import statement....178
usage guidelines....261
instance-name.inet.0 routing table....4
instance-type statement....275
usage guidelines....253
instances
routing, multiple....225
inter-area-prefix-export statement
OSPFv3....526
usage guidelines....485
inter-area-prefix-import statement
OSPFv3....527
usage guidelines....485
interface statement
IS-IS....277, 393
usage guidelines....327
multicast scoping....181
usage guidelines....117
multicast via static routes....180
neighbor discovery....687
usage guidelines....678
OSPF....277, 528
usage guidelines....454, 455
router discovery....668
routing options
usage guidelines....118
static routes
usage guidelines....58
interface-group (firewall filter match condition)....106
interface-routes statement....182
usage guidelines....116
interface-type statement....530
usage guidelines....455
interfaces descriptive text....274
Intermediate System-to-Intermediate System protocol *See* IS-IS
Internet Control Message Protocol router discovery *See* router discovery
ipsec-sa statement....531
BGP....840
OSPF
usage guidelines....488
usage guidelines....462, 745
ipv4-multicast statement
IS-IS....395
usage guidelines....338
ipv4-multicast-metric statement....395
usage guidelines....338
IPv6
addressing....12
representation....12
structure....13
types....12
benefits....10
EBGP link-local peering....786
header fields....11
ipv6-multicast statement
IS-IS....396
ipv6-multicast-metric statement....396
ipv6-unicast statement....397
usage guidelines....340
ipv6-unicast-metric statement....397
IS-IS
addresses....316
areas....341
authentication....325, 374, 410
CSNP....410
hello....411
PSNP....414
BFD....328, 376
checksum....336, 355
CLNS....356, 378
export BGP routes....356
pure ISO network....356
complete sequence number PDUs....317, 336, 379, 427
configuration statements....321
designated router....345, 420
disabling....342, 358, 381
IPv4 multicast topology....338
IPv4 unicast topology....338
IPv6 multicast topology....338
IPv6 routing....359
IPv6 unicast topology....340
enabling....358, 398
IPv6 routing....359
errored packets....427
graceful restart....350, 385
disable....350
hello
interval....343, 388
packet authentication....343, 387
packet authentication key....386
PDUs....317, 427
hold time....344, 390
hold-down timer
disabling....409
interfaces....393
IP fast reroute....363
IPv4 unicast topology....415
IPv6 unicast topology....397, 414
label-switched path....399

LDP synchronization.....336, 382
hold time.....391
level properties
global.....402
interfaces.....403
link protection
IS-IS.....365
link-protection statement.....404
link-state PDUs *See* IS-IS, LSPs
loop-free alternate routes.....363
loose authentication.....354, 405
LSPs.....317, 427
errored.....371
interval.....337, 405
lifetime.....347, 406
tracing.....427
mesh groups.....337, 371, 407
metrics.....344, 345, 421
IPv6.....397
multicast.....395, 396
normal.....408
traffic engineering.....425
wide.....346, 430
multicast reverse-path forwarding.....353
multicast topologies.....338, 395, 396
IPv4.....412
IPv6.....413
network PDUs.....315
no-eligible-backup statement.....411
node link protection.....365, 415
NSAP.....316
overloaded, marking router as.....348, 416, 481
packets *See* IS-IS, PDUs
padding.....355, 389
partial sequence number PDUs.....317, 427
PDUs.....317
point-to-point interface.....340, 418
policy, routing.....360, 382
preferences.....8, 347, 360, 383, 418
prefix limit.....347, 419
protocol data units *See* IS-IS, PDUs
route tagging.....318
routing domains.....403
routing instances
configure.....241
routing instances minimum configuration.....235
RSVP LSP backup paths.....366
SPF delay calculations.....427
standards supported.....318
topology.....426
tracing operations.....427
complete sequence number PDUs.....369
CSN PDUs.....369
description.....369
error PDUs.....369
graceful restart.....369
hello PDUs.....369
LDP synchronization.....369
LSP generation.....369
LSPs.....369
NSR synchronization.....369
partial sequence number PDUs.....369
policy processing.....369
protocol task processing.....369
protocol timer processing.....369
route information.....369
SPF delay calculations.....369
state transitions.....369
traffic engineering
lsp metrics.....524
traffic engineering
support.....344, 351, 353, 381, 423, 429
wide metrics.....346
isis statement.....398
usage guidelines.....324
ISO
addresses.....316
system identifier.....316
iso-vpn statement.....841
usage guidelines.....774

K

keep statement.....842
usage guidelines.....784
keepalive (tracing flag)
BGP.....875
keepalive messages.....718
kernel (tracing flag).....220
key-length statement.....704
usage guidelines.....701
key-pair statement.....704
usage guidelines.....701
keychain
BGP.....743

L

label-switched-path statement
IS-IS.....399
usage guidelines.....348
OSPF.....532
usage guidelines.....480
labeled-unicast statement.....843
usage guidelines.....742
last resort, route of generated routes *See* generated routes
Layer 2 VPN
routing instances
minimum configuration.....235

LDP
routing instances
configure multiple....245
minimum configuration....236
ldp-synchronization statement
IS-IS....400
usage guidelines....336
OSPF....533
usage guidelines....476
level statement
IS-IS
interfaces....403
protocol....402
usage guidelines....341
lifetime statement....669
usage guidelines....663
link-protection statement....404
OSPF....534
link-protection-statement
usage guidelines....365
link-state acknowledgment packets *See* OSPF, link-state acknowledgment packets
link-state advertisements *See* OSPF, link-state advertisements
link-state PDUs *See* IS-IS, LSPs
load sharing....10
local AS
BGP....756
local statement
OSPF....562
usage guidelines....488
local-address statement
BFD....154
usage guidelines....74
BGP....844
usage guidelines....746
local-as statement....845
usage guidelines....756
local-interface statement
BGP....847
usage guidelines....786
local-preference statement....848
usage guidelines....752
log (firewall filter action)....108
log-updown statement....849
BGP
usage guidelines....788
logging
routing protocol process....126, 192
logging, routing protocol process....126, 192
logical system....135
configuration statements....137
minimum configuration....138
overview....135
logical-systems statement....139
usage guidelines....138
loose-authentication-check statement
IS-IS....405
usage guidelines....354
loss-priority (firewall filter action)....108
LSAs *See* OSPF, link-state advertisements
lsp (tracing flag)....427
lsp-generation (tracing flag)....427
lsp-interval statement....405
usage guidelines....337
lsp-lifetime statement....406
usage guidelines....347
lsp-metric-into-summary statement....535
OSPF
usage guidelines....481
lsp-next-hop statement....183
usage guidelines....62
lsp-next-hop, static routes....183, 193
LSPs....317
MPLS, fate-sharing....166
See also IS-IS, LSPs, MPLS

M

managed-configuration statement....688
usage guidelines....679
manuals
comments on....xlii
martian addresses....45, 103, 184
martians statement....184
usage guidelines....103, 104
match conditions
firewall filters
overview....106
max-advertisement-interval statement....670, 688
ICMP
usage guidelines....663
neighbor discovery
usage guidelines....680
max-areas statement....406
usage guidelines....346
maximum-paths statement....185
usage guidelines....271
maximum-prefixes statement....186
usage guidelines....271
MD5 authentication....743
BGP....743
OSPF....458
md5 statement
OSPF....536
MED *See* BGP
med-igp-update-interval statement....187
usage guidelines....133
med-plus-igp statement....864
usage guidelines....754
members statement
usage guidelines....113

mesh groups....337, 407
mesh-group statement....407
usage guidelines....337
message-size statement....617
usage guidelines....596
metric statement
aggregate routes....188
usage guidelines....90
CLNS
usage guidelines....64
generated routes....188
IS-IS....408
usage guidelines....344
OSPF....537
usage guidelines....464, 488
qualified next hop....189
usage guidelines....58
static routes....188
usage guidelines....62, 67
metric-in statement
RIP....618
RIPng....648
usage guidelines....636
usage guidelines....596
metric-out statement
BGP....850
usage guidelines....747
RIP....619
usage guidelines....600
RIPng....649
usage guidelines....639
metric-type statement....539
usage guidelines....451
metrics
IS-IS....344, 345, 421
OSPF....464, 484, 558
RIP....600
RIPng....636, 639
static routes....67
min-advertisement-interval statement....671, 689
neighbor discovery
usage guidelines....680
usage guidelines....663
minimum-interval statement
BFD....154
usage guidelines....74
BGP....816
IS-IS....376
OSPF....507
usage guidelines....469
RIP....608
usage guidelines....588
minimum-receive-interval statement
BFD....154
usage guidelines....74
BFD (BGP)
usage guidelines....789
BGP....816
IS-IS....376
usage guidelines....328
OSPF....507
usage guidelines....469
RIP....608
usage guidelines....588
minimum-receive-ttl statement
BFD....154
BFD (static routes)
usage guidelines....74
MP-BGP....52, 767, 829
MPLS
ultimate-hop popping....827
mpls.0 routing table....4
MSDP
configuring multiple instances....246
MSDP routing instances, minimum configuration....236
mtu-discovery statement....852
BGP
usage guidelines....741
multiarea adjacency
OSPF....457
multicast
scoping....117
multicast statement....190
router discovery....672
usage guidelines....662
routing options
usage guidelines....117
multicast-rpf-routes statement....409
IS-IS
usage guidelines....353
OSPF
usage guidelines....481
multihop statement....853
usage guidelines....750
multipath statement....854
multiple active routes....6
multiplier statement
BFD....154
usage guidelines....74
BFD (BGP)
usage guidelines....789
BGP....816
IS-IS....376
usage guidelines....328
OSPF....507
usage guidelines....469
RIP....608
usage guidelines....588

multiprotocol BGP (MP-BGP).....767, 829
Multitopology Routing
BGP
configuring.....299
community statement.....304
filter-based forwarding.....300
OSPF
configuring.....292
overview.....287
static routes.....298
topologies
configuring.....291

N

neighbor discovery
autoconfiguration.....679
autonomous.....682
basics.....675
configuration statements.....31, 677
enabling.....678
frequency.....680
hop limit.....679
neighbor solicitation, frequency.....681
onlink.....681
preferred lifetime.....682
prefix information.....681
reachable time.....680
router advertisements.....678
router lifetime.....679
standards documents.....676
valid lifetime.....682
neighbor statement
BGP.....855
usage guidelines.....728, 734
OSPF.....540
usage guidelines.....455
RIP.....620
usage guidelines.....586
RIPng.....650
usage guidelines.....635
neighbor-discovery statement.....705
usage guidelines.....700
neighbors
BGP.....715
OSPF.....455
RIP.....586
RIPng.....635
network layer reachability information *See* BGP, NLRI
network link advertisements.....438
network PDUs.....315
network protocol data units *See* IS-IS, network PDUs
network service access point.....316
network-summary-export statement.....541
usage guidelines.....485
network-summary-import statement.....541
usage guidelines.....485
next-hop statement
CLNS
usage guidelines.....64
next-table statement
usage guidelines.....57
NLRI, BGP.....717
no-adaptation
BFD (BGP)
usage guidelines.....789
no-adaptation statement
BFD.....154
BFD (IS-IS)
usage guidelines.....328
BFD (static routes)
usage guidelines.....74
BGP.....816
IS-IS.....376
OSPF.....507
usage guidelines.....469
RIP.....608
usage guidelines.....588
no-adjacency-holddown statement.....409
usage guidelines.....355
no-advertise-peer-as statement
usage guidelines.....785
no-aggregator-id statement.....858
usage guidelines.....750
no-authentication-check statement.....410
usage guidelines.....325
no-autonomous statement
usage guidelines.....682
no-check-zero statement.....610
usage guidelines.....595
no-client-reflect statement.....859
usage guidelines.....761
no-csnp-authentication statement.....410
usage guidelines.....325
no-eligible-backup statement.....411
OSPF.....542
no-hello-authentication statement.....411
usage guidelines.....325
no-indirect-next-hop statement
usage guidelines.....128
no-install statement.....177
usage guidelines.....71
no-ipv4-multicast statement.....412
usage guidelines.....338
no-ipv4-routing statement.....412
no-ipv6-multicast statement.....413
usage guidelines.....338
no-ipv6-routing statement.....413
usage guidelines.....359
no-ipv6-unicast statement.....414

no-managed-configuration statement.....688
usage guidelines.....679
no-nssa-abr statement.....543
OSPF
usage guidelines.....453
no-on-link statement
usage guidelines.....681
no-other-stateful-configuration statement
usage guidelines.....679
no-prepend-global-as statement
usage guidelines.....756
no-psnp-authentication statement.....414
usage guidelines.....325
no-readvertise statement.....199
usage guidelines.....73
no-retain statement.....203
usage guidelines.....71
no-rfc-1583 statement.....544
usage guidelines.....453
no-unicast-topology statement.....415
usage guidelines.....338
no-validate statement.....860
no-vrf-advertise statement.....277
usage guidelines.....265
node-link-protection statement.....415
OSPF.....545
usage guidelines
IS-IS.....365
nonstop-routing statement.....191
usage guidelines.....129
normal (tracing flag).....220
neighbor discovery.....694, 697
RIPng.....655
notice (system logging severity level).....192
notice icons defined.....xl
NPDUs *See* IS-IS, network PDUs
NSAP.....316
nssa statement.....546
usage guidelines.....451

O

on-link statement.....690
usage guidelines.....681
open messages, BGP.....717
Open Shortest Path First *See* OSPF
options statement.....192
usage guidelines.....126
OSPF
adjacencies.....500
area border routers.....463
areas
configuring.....500
nonbackbone.....450
AS external link advertisements.....438
authentication.....458, 503
md5.....536
simple.....564
backbone.....450, 500
backup-spf-options statement.....504
bandwidth-based metrics.....505
configuring.....465
BFD.....469, 507
configuration statements.....444
configuring.....444
cost *See* OSPF, metrics
database description packets.....437
default route.....435
demand circuits.....456
designated router.....455, 462, 556
domain ID
configuring.....266
enabling.....277, 449, 454, 528, 547
error packets.....570
flood-reduction statement.....521
graceful restart.....477, 522
hello interval.....466, 523, 553
hello packets.....437
interface types.....530
label-switched path.....532
LDP synchronization.....476, 516
hold time.....524
link-protection statement.....534
link-state
acknowledgment packets.....438
advertisements.....438, 467, 559
flooding packets.....570
request packets.....438
update packets.....438
LSAs *See* OSPF, link-state advertisements
metric-type statement
usage guidelines.....451
metrics.....464, 484, 537, 558
traffic engineering.....569
multiarea adjacency
configuring.....457
NBMA networks.....455
neighbors.....455, 540, 546
network link advertisements.....438
no-eligible-backup statement.....542
node-link-protection statement.....545
nonbroadcast, multiaccess networks.....455
NSSAs.....512, 513
overload bit.....548
packets.....436, 438, 570
passive mode.....479, 550
passive traffic-engineering mode.....479
peer interfaces.....552
peer-interfaces.....488

policy, routing....484, 519, 525
network summaries....485
route install priority....486
preferences....8, 520, 554
prefix limit....462, 555
route cost *See* OSPF, metrics
route summarization....463, 501
route-type-community statement....561
usage guidelines....266
router dead interval....511
router identifier....113
router link advertisements....438
routing algorithm....433
routing instances, configure multiple....237, 247
sham link....488, 562
SPF....433, 570
standards documents....440
stub areas....450, 512, 513
summary link advertisements....438
tags
aggregate routes....93, 218
generated routes....101, 218
static routes....70, 218
timers....466
topological database....432
tracing operations....570
database description PDUs....495
demand circuit extensions....495
error PDUs....495
event....495
graceful restart....495
hello PDUs....495
LDP synchronization....495
link-state acknowledgement PDUs....495
link-state analysis PDUs....495
link-state PDUs....495
link-state request PDUs....495
link-state updates PDUs....495
NSR synchronization....495
packet dump....495
policy processing....495
protocol task processing....495
protocol timer processing....495
route information....495
SPF calculations....495
state transitions....495
traffic engineering features....574
traffic engineering support....484, 563
transmission delay....467, 576
transmit interval....577
virtual links....452, 579
ospf statement....547
usage guidelines....444, 449
ospf3 statement....547
OSPFv2
authentication, configuring....458
OSPFv3
authentication....462, 531
enabling....547
multiple address families
configuring....457
routing instances, configure multiple....238
other-stateful-configuration statement....690
usage guidelines....679
out-delay statement....861
usage guidelines....784
outbound-route-filter statement
BGP....862
overload statement
IS-IS....416
usage guidelines....348
OSPF....548
usage guidelines....481

P

p2mp-lsp-next-hop statement....193
usage guidelines....62
packet-dump (tracing flag)....570
packet-length (firewall filter match condition)....106
packets (tracing flag)
BGP....875
IS-IS....427
neighbor discovery....694, 697
OSPF....570
RIP....627
RIPng....655
parentheses, in syntax descriptions....xlii
parse (tracing flag)....220
partial sequence number PDUs *See* IS-IS, partial sequence number PDUs
passive statement....550
aggregate routes....142
usage guidelines....94
BGP....863
usage guidelines....746
generated routes....142
usage guidelines....102
IS-IS....417
usage guidelines....343, 345
OSPF
usage guidelines....479
static routes....142
usage guidelines....72
path attributes, BGP....715, 717, 750
path-selection statement....864
usage guidelines....754
PDUs *See* IS-IS, PDUs
peer-as statement....865
usage guidelines....731
peer-interface statement....552
usage guidelines....488

per-packet load balancing....120
physical interfaces, descriptive text....274
PIM
configuring multiple instances....250
PIM routing instances, minimum configuration....238
point-to-point statement....418
usage guidelines....340
policers
firewall filter action....108
policy (tracing flag)....220
neighbor discovery....694, 697
RIPng....655
policy statement
aggregate routes....194
usage guidelines....94
generated routes....194
usage guidelines....102
policy, routing
aggregate routes....94
BGP....828, 839
description....45
forwarding table....164
generated routes....102
IS-IS....360, 382
OSPF....484, 519, 525
network summaries....485
precedence....782
RIP....611, 616
RIPng....643, 647
routing instance....178
policy-based instance export, configuring....261
poll-interval statement....553
usage guidelines....455
port (firewall filter match condition)....106
ppm statement....195
usage guidelines....131
precedence (firewall filter match condition)....106
preference statement
aggregate routes....196
usage guidelines....90
BGP....866
usage guidelines....752
CLNS static routes
usage guidelines....64
generated routes....196
usage guidelines....99
IS-IS....418
usage guidelines....347
OSPF....554
usage guidelines....465
RIP....622
usage guidelines....599
RIPng....651
usage guidelines....638
static routes....196
usage guidelines....58, 62, 68
preferences
active routes....6, 7
aggregate routes....90, 196
generated routes....8
alternate preferences....6
default....8
generated routes....99
IS-IS....8, 347, 383, 418
modifying
with configuration statements....8
OSPF....520, 554
overview....6
RIP....8
static routes....8, 58, 62, 68, 196
tie-breaker preferences....6
preferred-lifetime statement....691
usage guidelines....682
prefix limit
IS-IS....347, 419
OSPF....462, 555
prefix statement....197
neighbor discovery....692
usage guidelines....678
usage guidelines....117
prefix-export-limit statement
IS-IS....419
usage guidelines....347
OSPF....555
usage guidelines....462
prefix-limit statement....867
usage guidelines....771, 781
primary routing tables....114
priority statement
IS-IS....420
usage guidelines....345
OSPF....556
usage guidelines....462
router discovery....673
protocol data units....317
See also IS-IS, PDUs
protocol-independent routing properties *See* aggregate routes
protocols
firewall filter match condition....106
match condition
firewall filters....106
protocols statement....278
psn (tracing flag)....427
PSNP IS-IS *See* IS-IS, partial sequence number PDUs

Q

qualified-next-hop statement....198
CLNS
usage guidelines....64
usage guidelines....58

R

rapid-runs statement
IS-IS....424
OSPF....565
reachable-time statement....692
usage guidelines....680
readvertise statement....199
usage guidelines....73
realm statement....557
usage guidelines....457
receive routes....57
receive statement
RIP....623
usage guidelines....597
RIPng....652
usage guidelines....636
static routes
usage guidelines....57
redirected routes....8
reference-bandwidth statement....558
IS-IS....421
usage guidelines....345, 464
regex-parse (tracing flag)....220
reject
firewall filters
action....108
reject option to static statement....215
usage guidelines....57
remove-private statement....869
usage guidelines....761
resolution statement....200
usage guidelines....127
resolution-ribs statement....200
usage guidelines....127
resolve statement....201
usage guidelines....74
resolve-vpn statement....870
usage guidelines....772
restart-duration statement....202
graceful restart
usage guidelines....124
IS-IS....385
usage guidelines....350
retain statement....203
usage guidelines....71, 84
retransmit-interval statement....559
usage guidelines....467
retransmit-timer statement....693
usage guidelines....681
rib statement
BGP....871
Multitopology Routing
static routes....305
route resolution....207
usage guidelines....127
routing tables....205
usage guidelines....52
rib-group statement....208
BGP....872
usage guidelines....767
IS-IS....422
OSPF....560
usage guidelines....487
RIP....624
usage guidelines....597
usage guidelines....116
rib-groups statement....209
usage guidelines....114
RIB-groups, static routes....63
RIP
authentication....587, 606
BFD....588
configuration statements....583
disable graceful restart....600
disabling address checks....605
enabling....585, 624
graceful restart....600, 612
groups....598
hold-down timer....615
metrics....618, 619
neighbors....586, 620
packets....582
policy, routing....611, 616
preferences....8, 599, 622
reserved fields....610
rib-group messages....597
rib-group statement
usage guidelines....597
route timeout....625
routing instances, configure multiple....251
standards documents....582
tracing operations
authentication....601
description....601
error PDUs....601
hold-down processing....601
NSR synchronization....601
policy processing....601
protocol task processing....601
protocol timer processing....601
request PDUs....601
route expiration processing....601
route information....601
state transitions....601
trigger updates....601
update PDUs....601
UDP, use of....581
update interval....629
update messages....596, 597, 617
rip statement....624
usage guidelines....585

RIPng
configuration statements....633
disable restart....639
enabling....653
graceful restart....639, 644
groups....637
holddown timer....646
metrics....636, 648, 649
neighbors....635, 650
overview....631
packets....632
policy, routing....638, 643, 647
preferences....638, 651
route timeout....653
standards documents....632
tracing operations
description....639
error PDUs....639
hold-down processing....639
NSR synchronization....639
policy processing....639
protocol task processing....639
protocol timer processing....639
request PDUs....639
route expiration processing....639
route information....639
state transitions....639
triggered updates....639
update PDUs....639
UDP, use of....631
update interval....657
ripng statement....653
usage guidelines....634
route
aggregate statement
usage guidelines....87
generate statement
usage guidelines....96
static statement....215
route (tracing flag)
neighbor discovery....694, 697
RIPng....655
routing....220
route distinguisher....210
route limit, configuring....271
paths....185
prefix....186
route of last resort *See* generated routes
route recording....210
route resolution....127
BGP....870
route statement
aggregate statement....143
generate statement....171
static statement
usage guidelines....54
route-distinguisher statement....281
usage guidelines....257
route-distinguisher-id statement....210
usage guidelines....125
route-record statement....210
usage guidelines....114
route-target statement....873
usage guidelines....779
route-timeout statement
RIP....625
usage guidelines....597
RIPng....653
usage guidelines....637
route-type-community statement....561
usage guidelines....266
router advertisements....662, 663, 670, 671
router discovery
configuration statements....661
designated router, configuring....673
router advertisements....659, 660, 662
router solicitations....659
server operation....659
server, enabling....662, 673
standards documents....660
tracing operations....663
router identifier....113, 211
router link advertisements....438
router-advertisement statement....693
router-discovery statement....673
usage guidelines....662
router-id statement....211
usage guidelines....113, 725
routes
aggregate *See* aggregate routes
contributing....87, 96
static *See* static routes
Routing Information Protocol *See* RIP
Routing Information Protocol next generation *See* RIPng
routing instances
BGP....234
configure....251
IS-IS....235
configuration example....242
LDP....236, 245
MSDP....246
multiple....225
router....242
second router....242
multiprotocol BGP-based multicast VPNs....237
OSPF....237, 247
configuration example....247
OSPFv3....238
PIM....238, 250

policy-based
auto-export configuration example.....261
instance-import configuration example.....263
RIP.....239, 251
router identifier.....210
routing protocol databases.....3
routing tables.....45
BGP, delays in exchanging routes.....784
creating.....52, 205
default.....52
default unicast.....52
export local routes.....116
flow routes.....52
group...114, 116, 165, 167, 174, 208, 209, 560, 872
import policy.....173
inet.0.....4
inet.1.....4
inet.2.....4, 52
inet.3.....4
inet6.0.....4, 52
instance-name.inet.0.....52
instance-name.inetflow.0.....52
instance-name.init.0.....4
mpls.0.....4
nonactive routes, exchanging with BGP.....783, 808
overview.....4
policy, routing.....178
primary.....114
secondary.....114
synchronizing.....5
routing-instance (firewall filter action).....108
routing-instances statement.....282
usage guidelines.....229, 240, 241
routing-options statement.....211
usage guidelines.....45
RSVP
preferences.....8

S

sample (firewall filter action).....108
scope statement.....212
usage guidelines.....117
scoping, multicast.....45, 117
secondary import and export policies, configure.....260
secondary routing tables.....114
secondary statement
OSPF interface.....561
usage guidelines.....457
Secure Neighbor Discovery
cryptographic addresses
configuring.....700
cryptographic-address statement.....703
enabling.....700
neighbor-discovery statement.....705
security-level statement.....707
timestamp statement.....708
secure statement.....706
security-level statement.....707
send statement
RIP.....626
usage guidelines.....597
RIPng.....654
usage guidelines.....636
sham-link statement.....562
usage guidelines.....488
sham-link-remote statement.....562
usage guidelines.....488
shortcuts statement
IS-IS.....423
usage guidelines.....351
OSPF.....563
usage guidelines.....481
simple-password statement.....564
source-address statement.....213
usage guidelines.....125
source-port (firewall filter match condition).....106
source-routing statement.....213
usage guidelines.....132
SPF.....315, 433
spf (tracing flag)
IS-IS.....427
OSPF.....570
spf-options statement
IS-IS.....424
usage guidelines.....349
OSPF.....565
usage guidelines.....478
SSM
groups, IGMP.....119
ssm-groups statement.....214
usage guidelines.....119
state (tracing flag)
neighbor discovery.....694, 697
RIPng.....655
routing protocols.....220
static options
static routes.....65
static routes.....45, 54, 215
BFD.....74, 154
Multitopology Routing.....298
preferences.....8
static statement.....215
usage guidelines.....54
stub areas.....450
stub statement.....567
usage guidelines.....450
summaries statement.....568
usage guidelines.....450
summary LSA.....438
support, technical *See* technical support

synchronizing routing information.....5
syntax conventions.....xli
syslog (firewall filter action).....108
syslog statement
routing options.....192
usage guidelines.....126
system ID *See* ISO, system identifier
system identifier *See* ISO, system identifier
system log messages
routing protocol process.....126, 192

T

tag statement.....218
aggregate routes
usage guidelines.....93
generated routes
usage guidelines.....101
static routes
usage guidelines.....70
task (tracing flag).....220
neighbor discovery.....694, 697
RIPng.....655
tcp-mss statement.....874
BGP
usage guidelines.....798
te-metric statement
IS-IS.....425
OSPF.....569
usage guidelines.....344, 484
technical support
contacting JTAC.....xlii
threshold statement.....219
BFD (BGP)
usage guidelines.....789
BGP.....816
IS-IS.....376
usage guidelines.....328
OSPF
usage guidelines.....469
RIP
usage guidelines.....588
usage guidelines.....119
tie-breaker preferences.....6
timer (tracing flag).....220, 655
neighbor discovery.....694, 697
timers
OSPF.....466
timestamp statement.....708
topologies statement
IS-IS.....426
Multitopology Routing.....306
usage guidelines.....291
topology statement
filter-based forwarding
Multitopology Routing.....308
usage guidelines.....300
Multitopology Routing.....309
OSPF.....310
OSPF interface.....311
usage guidelines.....292
topology-id statement
Multitopology Routing.....312
traceoptions
BFD.....151
traceoptions statement
BFD.....151
usage guidelines.....78
BGP.....875
description.....799
IS-IS.....427
description.....369
neighbor discovery.....694, 697
usage guidelines.....683
OSPF.....495, 570
RIP.....601, 627
RIPng.....639, 655
router discovery
usage guidelines.....663
routing protocols.....220
description.....129
Secure Neighbor Discovery.....709
tracing flags
all.....220
as-path.....875
auth.....627
config-internal.....220
csn.....427
damping.....875
error
IS-IS.....427
neighbor discovery.....694, 697
OSPF.....570
RIP.....627
RIPng.....655
expiration.....655
neighbor discovery.....694, 697
flash.....220
flooding.....570
general.....220
neighbor discovery.....694, 697
RIPng.....655
graceful restart
IS-IS.....427
OSPF.....570
hello
IS-IS.....427
holddown.....627, 655
neighbor discovery.....694, 697

keepalive
BGP....875
kernel....220
lsp....427
lsp-generation....427
normal....220
neighbor discovery....694, 697
RIPng....655
packet-dump....570
packets
BGP....875
IS-IS....427
neighbor discovery....694, 697
OSPF....570
RIP....627
RIPng....655
parse....220
policy....220
neighbor discovery....694, 697
RIPng....655
psn....427
regex-parse....220
route
neighbor discovery....694, 697
RIPng....655
routing....220
spf
IS-IS....427
OSPF....570
state
neighbor discovery....694, 697
RIPng....655
routing protocols....220
task....220
neighbor discovery....694, 697
RIPng....655
timer....220
neighbor discovery....694, 697
RIPng....655
trigger....627, 655
neighbor discovery....694, 697
update
neighbor discovery....694, 697
RIP....627
RIPng....655
tracing operations
BGP....799, 875
IS-IS....369, 427
neighbor discovery....694, 697
OSPF....495, 570
RIP....601, 627
RIPng....639, 655
router discovery....663
routing protocols....129, 220
traffic engineering database
OSPF support....574
traffic-engineering statement
IS-IS....429
usage guidelines....351
OSPF....574
usage guidelines....481
OSPF passive TE mode....575
usage guidelines....479
transit-delay statement....576
usage guidelines....467
transmit-interval statement....577
BFD....19, 154
BGP....816
IS-IS....376
OSPF
usage guidelines....469
trigger (tracing flag)....627, 655
neighbor discovery....694, 697
tunnel-type statement....222
usage guidelines....125
type statement....878
usage guidelines....731
type-7 statement....578
usage guidelines....451

U

unicast reverse path check....222
unicast RPF
example configuration....123
fail filters....123
unicast-reverse-path statement....222
usage guidelines....122
unnumbered Ethernet interfaces
as next-hop interface for static routes....58
configuration example....61
update (tracing flag)
neighbor discovery....694, 697
RIP....627
RIPng....655
update messages
BGP....717
update-interval statement
RIP....629
usage guidelines....597
RIPng....657
usage guidelines....637

V

valid-lifetime statement....698
usage guidelines....682
validation statement
usage guidelines....105

version statement
BFD....154
usage guidelines....74
BFD (BGP)
usage guidelines....789
BGP....816
IS-IS....376
usage guidelines....328
OSPF....507
usage guidelines....469
RIP....608
usage guidelines....588
virtual-link statement....579
usage guidelines....452
VPLS
routing instances
minimum configuration....239
vpn-apply-export statement....878
usage guidelines....789
VRF export policy....878
VRF table label, configuring....265
VRF target, configuring....265
vrf-export statement....283
usage guidelines....260
vrf-import statement....283
usage guidelines....260
vrf-table-label statement....284
usage guidelines....265
vrf-target statement....284
usage guidelines....265

W

warning (system logging severity level)....192
wide-metrics-only statement....430
usage guidelines....346

Index of Statements and Commands

A

accept-remote-nexthop statement....804
accepted-prefix-limit statement....805
access-profile statement
    routing instances....273
active statement
    aggregate routes....142
    generated routes....142
    static routes....142
address statement....665
advertise statement....666
advertise-external statement....807
advertise-inactive statement....808
advertise-peer-as statement....809
aggregate statement....143
aggregate-label statement....810
aggregator statement....146
allow statement....811
any-sender statement
    RIP....605
area statement....500
area-range statement....501
as-override statement....812
as-path statement....146
authentication-algorithm statement
    BGP....813
authentication-key statement
    BGP....814
    IS-IS....374
    RIP....606
authentication-key-chain statement....815
authentication-type statement
    IS-IS....375
    RIP....607
auto-export statement....148
autonomous statement....685
autonomous-system statement....149

B

backup-spf-options statement
    OSPF....504
bandwidth-based-metrics statement....505
bfd statement....151
bfd-liveness-detection statement
    BGP....816
    IS-IS....376
    OSPF....507
    RIP....608
    static routes....154
bgp statement....820
bgp-orf-cisco-mode statement....821
bmp statement....822
brief statement....158
broadcast statement....666

C

check-zero statement....610
checksum statement....378
clns-routing statement
    IS-IS....378
cluster statement....823
color statement
    aggregate routes....196
    generated routes....196
    static routes....196
community statement
    aggregate routes....159
    generated routes....159
    Multitopology Routing....304
    static routes....159
confederation statement....161
credibility-protocol-preference
    traffic engineering
        IS-IS....429
cryptographic-address statement....703
csnp-interval statement....379
current-hop-limit statement....686

D

damping statement....824
dead-interval statement....511
default-lifetime statement....686
default-lsa statement....512
default-metric statement....513

defaults statement
aggregate statement....143
generate statement....171
static statement....215
delay statement
IS-IS....424
OSPF....565
demand-circuit statement....514
description statement....274, 825
destination-networks statement....161
detection-time statement
BFD....154
BGP....816
IS-IS....376
disable statement
BGP....826
IS-IS....381
graceful restart....385
LDP synchronization....382
OSPF....517
LDP synchronization....516
discard statement
aggregate routes....163
generated routes....163
domain-id statement....518
domain-vpn-tag statement....518
dynamic-tunnels statement....164

E

enable statement
routing options....180
explicit-null statement....827
export statement
BGP....828
forwarding table....164
IS-IS....382
OSPF....519
RIP....611
RIPng....643
export-rib statement....165
external-preference statement
IS-IS....383
OSPF....520

F

family statement
BGP....829
IS-IS....384
fate-sharing statement....166
filter statement....167
flood-reduction statement....521
flow statement....168, 829
forwarding-cache statement....169
forwarding-table statement....169
full statement....158

G

generate statement....171
graceful-restart statement
BGP....833
IS-IS....385
OSPF....522
RIP....612
RIPng....644
group statement
BGP....834
RIP....613
RIPng....645

H

hello-authentication-key statement....386
hello-authentication-type statement....387
hello-interval statement
IS-IS....388
OSPF....523
hello-padding statement....389
helper-disable statement
IS-IS....385
hold-time statement
BGP....837
IS-IS....390
LDP synchronization....391
OSPF
LDP synchronization....524
holddown statement
IS-IS....424
OSPF....565
RIP....615
RIPng....646
holddown-interval statement
BFD
static routes....154

I

idle-after-switch-over statement....838
ignore statement....666, 667
ignore-attached-bit statement....391
ignore-lsp-metrics statement
IS-IS....392
OSPF....524
import statement
BGP....839
OSPF....525
RIP....616
RIPng....647
route resolution....173
import-policy statement....173

import-rib statement....174
include-mp-next-hop statement....840
independent-domain statement....175
indirect-next-hop statement....176
ineligible statement....667, 673
input statement....176
install statement....177
instance-export statement....178
instance-import statement....178
instance-type statement....275
inter-area-prefix-export statement
OSPFv3....526
inter-area-prefix-import statement
OSPFv3....527
interface statement
IS-IS....277, 393
multicast scoping....181
multicast via static routes....180
neighbor discovery....687
OSPF....277, 528
interface-routes statement....182
interface-type statement....530
ipsec-sa statement....531
BGP....840
ipv4-multicast statement
IS-IS....395
ipv4-multicast-metric statement....395
ipv6-multicast statement
IS-IS....396
ipv6-multicast-metric statement....396
ipv6-unicast statement....397
ipv6-unicast-metric statement....397
isis statement....398
iso-vpn statement....841

K

keep statement....842
key-length statement....704
key-pair statement....704

L

label-switched-path statement
IS-IS....399
OSPF....532
labeled-unicast statement....843
ldp-synchronization statement
IS-IS....400
OSPF....533
level statement
IS-IS
interfaces....403
protocol....402
lifetime statement....669
link-protection statement....404
OSPF....534
local statement
OSPF....562
local-address statement
BFD....154
BGP....844
local-as statement....845
local-interface statement
BGP....847
local-preference statement....848
log-updown statement....849
logical-systems statement....139
loose-authentication-check statement
IS-IS....405
lsp-interval statement....405
lsp-lifetime statement....406
lsp-metric-into-summary statement....535
lsp-next-hop statement....183

M

managed-configuration statement....688
martians statement....184
max-advertisement-interval statement....670, 688
max-areas statement....406
maximum-paths statement....185
maximum-prefixes statement....186
md5 statement
OSPF....536
med-igp-update-interval statement....187
med-plus-igp statement....864
mesh-group statement....407
message-size statement....617
metric statement
aggregate routes....188
generated routes....188
IS-IS....408
OSPF....537
qualified next hop....189
static routes....188
metric-in statement
RIP....618
RIPng....648
metric-out statement
BGP....850
RIP....619
RIPng....649
metric-type statement....539
min-advertisement-interval statement....671, 689
minimum-interval statement
BFD....154
BGP....816
IS-IS....376
OSPF....507
RIP....608

minimum-receive-interval statement
BFD....154
BGP....816
IS-IS....376
OSPF....507
RIP....608
minimum-receive-ttl statement
BFD....154
mtu-discovery statement....852
multicast statement....190
router discovery....672
multicast-rpf-routes statement....409
multihop statement....853
multipath statement....854
multiplier statement
BFD....154
BGP....816
IS-IS....376
OSPF....507
RIP....608

N

neighbor statement
BGP....855
OSPF....540
RIP....620
RIPng....650
neighbor-discovery statement....705
network-summary-export statement....541
network-summary-import statement....541
no-adaptation statement
BFD....154
BGP....816
IS-IS....376
OSPF....507
RIP....608
no-adjacency-holddown statement....409
no-aggregator-id statement....858
no-authentication-check statement....410
no-check-zero statement....610
no-client-reflect statement....859
no-csnp-authentication statement....410
no-eligible-backup statement....411
OSPF....542
no-hello-authentication statement....411
no-install statement....177
no-ipv4-multicast statement....412
no-ipv4-routing statement....412
no-ipv6-multicast statement....413
no-ipv6-routing statement....413
no-ipv6-unicast statement....414
no-managed-configuration statement....688
no-nssa-abr statement....543
no-psnp-authentication statement....414
no-readvertise statement....199
no-retain statement....203
no-rfc-1583 statement....544
no-unicast-topology statement....415
no-validate statement....860
no-vrf-advertise statement....277
node-link-protection statement....415
OSPF....545
nonstop-routing statement....191
nssa statement....546

O

on-link statement....690
options statement....192
ospf statement....547
ospf3 statement....547
other-stateful-configuration statement....690
out-delay statement....861
outbound-route-filter statement
BGP....862
overload statement
IS-IS....416
OSPF....548

P

p2mp-lsp-next-hop statement....193
passive statement....550
aggregate routes....142
BGP....863
generated routes....142
IS-IS....417
static routes....142
path-selection statement....864
peer-as statement....865
peer-interface statement....552
point-to-point statement....418
policy statement
aggregate routes....194
generated routes....194
poll-interval statement....553
ppm statement....195
preference statement
aggregate routes....196
BGP....866
generated routes....196
IS-IS....418
OSPF....554
RIP....622
RIPng....651
static routes....196
preferred-lifetime statement....691
prefix statement....197
neighbor discovery....692

prefix-export-limit statement
IS-IS....419
OSPF....555
prefix-limit statement....867
priority statement
IS-IS....420
OSPF....556
router discovery....673
protocols statement....278

Q

qualified-next-hop statement....198

R

rapid-runs statement
IS-IS....424
OSPF....565
reachable-time statement....692
readvertise statement....199
realm statement....557
receive statement
RIP....623
RIPng....652
reference-bandwidth statement....558
IS-IS....421
remove-private statement....869
resolution statement....200
resolution-ribs statement....200
resolve statement....201
resolve-vpn statement....870
restart-duration statement....202
IS-IS....385
retain statement....203
retransmit-interval statement....559
retransmit-timer statement....693
rib statement
BGP....871
Multitopology Routing
static routes....305
route resolution....207
routing tables....205
rib-group statement....208
BGP....872
IS-IS....422
OSPF....560
RIP....624
rib-groups statement....209
rip statement....624
ripng statement....653
route statement
aggregate statement....143
generate statement....171
route-distinguisher statement....281
route-distinguisher-id statement....210
route-record statement....210
route-target statement....873
route-timeout statement
RIP....625
RIPng....653
route-type-community statement....561
router-advertisement statement....693
router-discovery statement....673
router-id statement....211
routing-instances statement....282
routing-options statement....211

S

scope statement....212
secondary statement
OSPF interface....561
secure statement....706
security-level statement....707
send statement
RIP....626
RIPng....654
sham-link statement....562
sham-link-remote statement....562
shortcuts statement
IS-IS....423
OSPF....563
simple-password statement....564
source-address statement....213
source-routing statement....213
spf-options statement
IS-IS....424
OSPF....565
ssm-groups statement....214
static statement....215
stub statement....567
summaries statement....568
syslog statement
routing options....192

T

tag statement....218
tcp-mss statement....874
te-metric statement
IS-IS....425
OSPF....569
threshold statement....219
BGP....816
IS-IS....376
timestamp statement....708
topologies statement
IS-IS....426
Multitopology Routing....306

topology statement
filter-based forwarding
Multitopology Routing.....308
Multitopology Routing.....309
OSPF.....310
OSPF interface.....311
topology-id statement
Multitopology Routing.....312
traceoptions statement
BGP.....875
IS-IS.....427
neighbor discovery.....694, 697
OSPF.....495, 570
RIP.....601, 627
RIPng.....639, 655
routing protocols.....220
Secure Neighbor Discovery.....709
traffic-engineering statement
IS-IS.....429
OSPF.....574
OSPF passive TE mode.....575
transit-delay statement.....576
transmit-interval statement.....577
BFD.....19, 154
BGP.....816
IS-IS.....376
tunnel-type statement.....222
type statement.....878
type-7 statement.....578

U

unicast-reverse-path statement.....222
update-interval statement
RIP.....629
RIPng.....657

V

valid-lifetime statement.....698
version statement
BFD.....154
BGP.....816
IS-IS.....376
OSPF.....507
RIP.....608
virtual-link statement.....579
vpn-apply-export statement.....878
vrf-export statement.....283
vrf-import statement.....283
vrf-table-label statement.....284
vrf-target statement.....284

W

wide-metrics-only statement.....430