



**Huawei AR150&200 Series Enterprise Routers
V200R002C00**

Configuration Guide - IP Routing

Issue 02
Date 2012-03-30

Copyright © Huawei Technologies Co., Ltd. 2012. All rights reserved.

No part of this document may be reproduced or transmitted in any form or by any means without prior written consent of Huawei Technologies Co., Ltd.

Trademarks and Permissions



HUAWEI and other Huawei trademarks are trademarks of Huawei Technologies Co., Ltd.

All other trademarks and trade names mentioned in this document are the property of their respective holders.

Notice

The purchased products, services and features are stipulated by the contract made between Huawei and the customer. All or part of the products, services and features described in this document may not be within the purchase scope or the usage scope. Unless otherwise specified in the contract, all statements, information, and recommendations in this document are provided "AS IS" without warranties, guarantees or representations of any kind, either express or implied.

The information in this document is subject to change without notice. Every effort has been made in the preparation of this document to ensure accuracy of the contents, but all statements, information, and recommendations in this document do not constitute the warranty of any kind, express or implied.

Huawei Technologies Co., Ltd.

Address: Huawei Industrial Base
Bantian, Longgang
Shenzhen 518129
People's Republic of China

Website: <http://www.huawei.com>

Email: support@huawei.com

About This Document

Intended Audience

This document provides the basic concepts, configuration procedures, and configuration examples for different application scenarios of the AR150/200. Topics covered include static routes, routing protocols (RIP, OSPF, and IS-IS), and routing policies.






This document describes how to configure the IP routing features.

This document is intended for:

- Data configuration engineers
- Commissioning engineers
- Network monitoring engineers
- System maintenance engineers

Symbol Conventions

The symbols that may be found in this document are defined as follows.

Symbol	Description
 DANGER	Indicates a hazard with a high level of risk, which if not avoided, results in death or serious injury.
 WARNING	Indicates a hazard with a medium or low level of risk, which if not avoided, could result in minor or moderate injury.
 CAUTION	Indicates a potentially hazardous situation, which if not avoided, could result in equipment damage, data loss, performance degradation, or unexpected results.
 TIP	Indicates a tip that may help you solve a problem or save time.
 NOTE	Provides additional information to emphasize or supplement important points of the main text.

Command Conventions

The command conventions that may be found in this document are defined as follows.

Convention	Description
Boldface	The keywords of a command line are in boldface .
<i>Italic</i>	Command arguments are in <i>italics</i> .
[]	Items (keywords or arguments) in brackets [] are optional.
{ x y ... }	Optional items are grouped in braces and separated by vertical bars. One item is selected.
[x y ...]	Optional items are grouped in brackets and separated by vertical bars. One item is selected or no item is selected.
{ x y ... }*	Optional items are grouped in braces and separated by vertical bars. A minimum of one item or a maximum of all items can be selected.
[x y ...]*	Optional items are grouped in brackets and separated by vertical bars. Several items or no item can be selected.
&<1-n>	The parameter before the & sign can be repeated 1 to n times.
#	A line starting with the # sign is comments.

Interface Numbering Conventions

Interface numbers used in this manual are examples. In device configuration, use the existing interface numbers on devices.

Change History

Updates between document issues are cumulative. Therefore, the latest document issue contains all updates made in previous issues.

Changes in Issue 02 (2012-03-30)

Based on issue 01 (2011-12-30), the document is updated as follows:

The following information is modified:

- [7.3.2 Creating IPv4 IS-IS Processes](#)

Changes in Issue 01 (2011-12-30)

Initial commercial release.

Contents

About This Document.....	ii
1 IP Routing Basic Configuration.....	1
1.1 Routing Management.....	2
1.1.1 Displaying of the Routing Table.....	2
1.1.2 Displaying of the Routing Management Module.....	2
2 IP Static Route Configuration.....	4
2.1 Static Route.....	5
2.2 Static Routing Features Supported by the AR150/200.....	5
2.3 Configuring an IPv4 Static Route.....	6
2.3.1 Establishing the Configuration Task.....	6
2.3.2 Configuring an IPv4 Static Route on the Public Network.....	7
2.3.3 (Optional) Setting the Default Preference for IPv4 Static Routes.....	8
2.3.4 (Optional) Configuring Static Route Selection Based on Relay Depth.....	8
2.3.5 (Optional) Configuring Permanent Advertisement of IPv4 Static Routes.....	9
2.3.6 Checking the Configuration.....	10
2.4 Configuring an IPv6 Static Route.....	10
2.4.1 Establishing the Configuration Task.....	10
2.4.2 Configuring an IPv6 Static Route on the Public Network.....	11
2.4.3 (Optional) Setting the Default Preference for IPv6 Static Routes.....	11
2.4.4 Checking the Configuration.....	12
2.5 Configuring BFD for IPv4 Static Routes on the Public Network.....	12
2.5.1 Establishing the Configuration Task.....	12
2.5.2 Configuring an IPv4 Static Route on the Public Network.....	13
2.5.3 Configuring a BFD Session.....	14
2.5.4 Binding a Static Route to a BFD Session.....	15
2.5.5 Checking the Configuration.....	15
2.6 Configuring NQA for IPv4 Static Routes.....	16
2.6.1 Establishing the Configuration Task.....	16
2.6.2 Configuring an ICMP Type NQA Test Instance.....	17
2.6.3 Binding an IPv4 Static Route to an NQA Test Instance.....	18
2.6.4 Checking the Configuration.....	19
2.7 Configuration Examples.....	20

2.7.1 Example for Configuring IPv4 Static Routes.....	20
2.7.2 Example for Configuring IPv6 Static Routes.....	24
3 RIP Configuration.....	28
3.1 Overview of RIP.....	30
3.2 RIP Features Supported by the AR150/200.....	31
3.3 Configuring Basic RIP Functions.....	31
3.3.1 Establishing the Configuration Task.....	31
3.3.2 Enabling RIP.....	31
3.3.3 Enabling RIP on the Specified Network Segment.....	32
3.3.4 Configuring RIP Version Number.....	33
3.3.5 Checking the Configuration.....	34
3.4 Configuring RIP Route Attributes.....	34
3.4.1 Establishing the Configuration Task.....	34
3.4.2 Configuring Additional Metrics of an Interface.....	35
3.4.3 Configuring RIP Preference.....	36
3.4.4 Setting the Maximum Number of Equal-Cost Routes.....	36
3.4.5 Checking the Configuration.....	37
3.5 Controlling the Advertising of RIP Routing Information.....	37
3.5.1 Establishing the Configuration Task.....	37
3.5.2 Configuring RIP to Advertise Default Routes.....	38
3.5.3 Disabling an Interface from Sending Update Packets.....	38
3.5.4 Configuring RIP to Import External Routes.....	39
3.5.5 Checking the Configuration.....	40
3.6 Controlling the Receiving of RIP Routing Information.....	41
3.6.1 Establishing the Configuration Task.....	41
3.6.2 Disabling an Interface from Receiving RIP Update Packets.....	41
3.6.3 Disabling RIP from Receiving Host Routes.....	42
3.6.4 Configuring RIP to Filter the Received Routes.....	43
3.6.5 Checking the Configuration.....	44
3.7 Configuring RIP-2 Features.....	44
3.7.1 Establishing the Configuration Task.....	44
3.7.2 Configuring RIP-2 Route Summarization.....	45
3.7.3 Configuring Packet Authentication of RIP-2.....	46
3.7.4 Checking the Configuration.....	47
3.8 Optimizing a RIP Network.....	47
3.8.1 Establishing the Configuration Task.....	48
3.8.2 Configuring RIP Timers.....	48
3.8.3 Setting the Interval for Sending Packets and the Maximum Number of the Sent Packets.....	49
3.8.4 Configuring Split Horizon and Poison Reverse.....	50
3.8.5 Enabling replay-protect Function.....	51
3.8.6 Configuring RIP to Check the Validity of Update Packets.....	52
3.8.7 Configuring RIP Neighbors.....	53

3.8.8 Checking the Configuration.....	53
3.9 Configuring RIP GR.....	54
3.9.1 Establishing the Configuration Task.....	54
3.9.2 Enabling RIP GR.....	55
3.9.3 Checking the Configuration.....	55
3.10 Configuring BFD for RIP.....	55
3.11 Configuring Static BFD for RIP.....	58
3.12 Configuring the Network Management Function in RIP.....	61
3.12.1 Establishing the Configuration Task.....	61
3.12.2 Binding RIP to MIBs.....	61
3.12.3 Checking the Configuration.....	62
3.13 Maintaining RIP.....	62
3.13.1 Resetting RIP.....	62
3.13.2 Clearing RIP.....	63
3.14 Configuration Examples.....	63
3.14.1 Example for Configuring RIP Version.....	63
4 RIPng Configuration.....	67
4.1 RIPng Overview.....	68
4.2 RIPng Features Supported by the AR150/200.....	69
4.3 Configuring Basic RIPng Functions.....	69
4.3.1 Establishing the Configuration Task.....	69
4.3.2 Enabling RIPng and Entering the RIPng View.....	70
4.3.3 Enabling RIPng in the Interface View.....	70
4.3.4 Checking the Configuration.....	71
4.4 Configuring RIPng Route Attributes.....	71
4.4.1 Establishing the Configuration Task.....	71
4.4.2 Configuring the RIPng Preference.....	72
4.4.3 Configuring Additional Metrics of an Interface.....	73
4.4.4 Configuring the Maximum Number of Equal-Cost Routes.....	74
4.4.5 Checking the Configuration.....	74
4.5 Controlling the Advertising of RIPng Routing Information.....	74
4.5.1 Establishing the Configuration Task.....	75
4.5.2 Configuring RIPng Route Summarization.....	75
4.5.3 Configuring RIPng to Advertise the Default Routes.....	76
4.5.4 Configuring the Default Cost for External Routes Imported by RIPng.....	76
4.5.5 Configuring RIPng to Import External Routes.....	77
4.5.6 Checking the Configuration.....	78
4.6 Controlling the Receiving of RIPng Routing Information.....	78
4.6.1 Establishing the Configuration Task.....	78
4.6.2 Configuring RIPng to Filter the Received Routes.....	79
4.6.3 Checking the Configuration.....	79
4.7 Optimizing a RIPng Network.....	80

4.7.1 Establishing the Configuration Task.....	80
4.7.2 Configuring RIPng Timers.....	81
4.7.3 Setting the Interval for Sending Update Packets and the Maximum Number of Packets Sent Each Time	81
4.7.4 Configuring Split Horizon and Poison Reverse.....	82
4.7.5 Enabling the Zero Field Check for RIPng Packets.....	82
4.7.6 Checking the Configuration.....	83
4.8 Maintaining RIPng.....	83
4.8.1 Clearing RIPng.....	84
4.9 Configuration Examples.....	84
4.9.1 Example for Configuring Basic RIPng Functions.....	84
5 OSPF Configuration.....	87
5.1 OSPF Overview.....	89
5.2 OSPF Features Supported by the AR150/200.....	92
5.3 Configuring Basic OSPF Functions.....	95
5.3.1 Establishing the Configuration Task.....	95
5.3.2 Enabling OSPF.....	96
5.3.3 (Optional) Creating OSPF Virtual Links.....	97
5.3.4 (Optional) Configuring a Route Selection Rule on the router.....	98
5.3.5 (Optional) Setting the OSPF Priority.....	99
5.3.6 (Optional) Restricting the Flooding of LSA Update Packets.....	100
5.3.7 (Optional) Configuring the Maximum Number of Packet Retransmission Attempts.....	100
5.3.8 (Optional) Setting an Interval at Which an LSA Packet Is Retransmitted to the Neighboring router	101
5.3.9 (Optional) Configuring an Interface to Fill in a DD Packet with the Interface MTU.....	102
5.3.10 Checking the Configuration.....	103
5.4 Configuring OSPF on the NBMA or P2MP Network.....	103
5.4.1 Establishing the Configuration Task.....	103
5.4.2 Configuring Network Types for OSPF Interfaces.....	105
5.4.3 Configuring NBMA Network Attributes.....	106
5.4.4 Configuring P2MP Network Attributes.....	107
5.4.5 Checking the Configuration.....	108
5.5 Configuring an OSPF Route Selection Rule.....	108
5.5.1 Establishing the Configuration Task.....	109
5.5.2 Setting the Interface Cost.....	109
5.5.3 Configuring Equal-Cost Routes.....	110
5.5.4 Configuring a Stub Router.....	112
5.5.5 Suppressing an Interface from Receiving and Sending OSPF Packets.....	112
5.5.6 Checking the Configuration.....	113
5.6 Controlling OSPF Routing Information.....	113
5.6.1 Establishing the Configuration Task.....	114
5.6.2 Configuring OSPF to Import External Routes.....	114
5.6.3 Configuring OSPF to Import a Default Route.....	115

5.6.4 Configuring Route Summarization.....	117
5.6.5 Configuring OSPF to Filter Routes Received by OSPF.....	118
5.6.6 Configuring the router to Filter LSAs to Be Sent.....	118
5.6.7 (Optional) Configuring OSPF to Filter LSAs in an Area.....	119
5.6.8 (Optional) Enabling the Mesh-Group Function.....	120
5.6.9 Setting the Maximum Number of External LSAs in the LSDB.....	120
5.6.10 Checking the Configuration.....	121
5.7 Configuring an OSPF Stub Area.....	121
5.8 Configuring an NSSA.....	123
5.9 Configuring BFD for OSPF.....	126
5.9.1 Establishing the Configuration Task.....	126
5.9.2 Configuring BFD for OSPF in a Specified Process.....	127
5.9.3 Configuring BFD for OSPF on a Specified Interface.....	129
5.9.4 Checking the Configuration.....	130
5.10 Configuring OSPF GR.....	131
5.10.1 Establishing the Configuration Task.....	131
5.10.2 Enabling OSPF GR.....	131
5.10.3 (Optional) Configuring the GR Session Parameters on the Restarter.....	132
5.10.4 (Optional) Configuring GR Session Parameters on the Helper.....	133
5.10.5 Checking the Configuration.....	133
5.11 Improving Security of an OSPF Network.....	133
5.11.1 Establishing the Configuration Task.....	134
5.11.2 Configuring the OSPF GTSM Functions.....	134
5.11.3 Configuring the Authentication Mode.....	136
5.11.4 Checking the Configuration.....	137
5.12 Configuring the Network Management Function of OSPF.....	138
5.12.1 Establishing the Configuration Task.....	138
5.12.2 Configuring OSPF MIB Binding.....	138
5.12.3 Configuring OSPF Trap.....	139
5.12.4 Configuring OSPF Log.....	139
5.12.5 Checking the Configuration.....	140
5.13 Maintaining OSPF.....	140
5.13.1 Resetting OSPF.....	140
5.13.2 Clearing OSPF.....	141
5.14 Configuration Examples.....	141
5.14.1 Example for Configuring Basic OSPF Functions.....	142
5.14.2 Example for Configuring DR Election of OSPF.....	147
5.14.3 Example for Configuring OSPF Stub Areas.....	151
6 OSPFv3 Configuration.....	156
6.1 OSPFv3 Overview.....	158
6.2 OSPFv3 Features Supported by AR150/200.....	158
6.3 Configuring Basic OSPFv3 Functions.....	159

6.3.1 Establishing the Configuration Task.....	159
6.3.2 Enabling OSPFv3.....	159
6.3.3 Enabling OSPFv3 on an Interface.....	160
6.3.4 Entering the OSPFv3 Area View.....	161
6.3.5 Checking the Configuration.....	162
6.4 Establishing or Maintaining OSPFv3 Neighbor Relationship.....	162
6.4.1 Establishing the Configuration Task.....	162
6.4.2 Configuring the Interval for Sending Hello Packets.....	163
6.4.3 Configuring Dead Time of Neighbor Relationship.....	164
6.4.4 Configuring the Interval for Retransmitting LSAs to Neighboring	164
6.4.5 Configuring the Delay for Transmitting LSAs on the Interface.....	165
6.4.6 Checking the Configuration.....	166
6.5 Configuring OSPFv3 Areas.....	166
6.5.1 Establishing the Configuration Task.....	166
6.5.2 Configuring OSPFv3 Stub Areas.....	167
6.5.3 Configuring OSPFv3 Virtual Links.....	167
6.5.4 Checking the Configuration.....	168
6.6 Configuring OSPFv3 NSSA Areas.....	169
6.6.1 Establishing the Configuration Task.....	169
6.6.2 Defining the Current Area to Be an NSSA Area.....	169
6.6.3 Checking the Configuration.....	170
6.7 Configuring OSPFv3 Route Attributes.....	171
6.7.1 Establishing the Configuration Task.....	171
6.7.2 Setting the Cost of the OSPFv3 Interface.....	171
6.7.3 Setting the Maximum Number of OSPFv3 Equal-Cost Routes.....	172
6.7.4 Checking the Configuration.....	172
6.8 Controlling OSPFv3 Routing Information.....	173
6.8.1 Establishing the Configuration Task.....	173
6.8.2 Configuring OSPFv3 Route Aggregation.....	174
6.8.3 Configuring OSPFv3 to Filter the Received Routes.....	175
6.8.4 Configuring OSPFv3 to Import External Routes.....	176
6.8.5 Checking the Configuration.....	177
6.9 Optimizing an OSPFv3 Network.....	177
6.9.1 Establishing the Configuration Task.....	177
6.9.2 Configuring the SPF Timer.....	178
6.9.3 Setting the Interval for Receiving LSAs.....	179
6.9.4 Configuring an Intelligent Timer for Generating LSAs.....	180
6.9.5 Suppressing an Interface from Sending and Receiving OSPFv3 Packets.....	180
6.9.6 Configuring DR Priority of an Interface.....	181
6.9.7 Configuring Stub Routers.....	182
6.9.8 Ignoring MTU Check on DD Packets.....	183
6.9.9 Checking the Configuration.....	183

6.10 Configuration OSPFv3 GR.....	184
6.10.1 Establishing the Configuration Task.....	184
6.10.2 Enabling OSPFv3 GR.....	184
6.10.3 Enabling the Helper of OSPFv3 GR.....	185
6.10.4 Check the Configuration.....	186
6.11 Configuring the Network Management Function of OSPFv3.....	186
6.11.1 Establishing the Configuration Task.....	186
6.11.2 Configuring OSPFv3 MIB Binding.....	186
6.11.3 Configuring OSPFv3 Trap.....	187
6.11.4 Check the Configuration.....	187
6.12 Maintaining OSPFv3.....	188
6.12.1 Resetting OSPFv3.....	188
6.13 Configuration Examples.....	188
6.13.1 Example for Configuring OSPFv3 Areas.....	188
6.13.2 Example for Configuring OSPFv3 DR Election.....	193
7 IS-IS Configuration.....	198
7.1 Basic Concepts of IS-IS.....	200
7.2 IS-IS Features Supported by the AR150/200.....	201
7.3 Configuring Basic IPv4 IS-IS Functions.....	206
7.3.1 Establishing the Configuration Task.....	207
7.3.2 Creating IPv4 IS-IS Processes.....	207
7.3.3 Configuring IPv4 IS-IS Interfaces.....	209
7.3.4 (Optional) Configuring the IPv4 IS-IS Interfaces.....	211
7.3.5 (Optional) Configuring IPv4 IS-IS Attributes for Interfaces on Different Types of Networks.....	214
7.3.6 Checking the Configuration.....	216
7.4 Establishing or Maintaining IS-IS Neighbor Relationships or Adjacencies.....	217
7.4.1 Establishing the Configuration Task.....	217
7.4.2 Configuring IS-IS Timers for Packets.....	217
7.4.3 Configuring LSP Parameters.....	220
7.4.4 Checking the Configuration.....	223
7.5 Configuring IPv4 IS-IS Route Selection.....	223
7.5.1 Establishing the Configuration Task.....	223
7.5.2 Configuring IPv4 IS-IS Route Leaking.....	224
7.5.3 Configuring Principles for Using Equal-Cost IPv4 IS-IS Routes.....	226
7.5.4 Filtering IPv4 IS-IS Routes.....	227
7.5.5 Configuring an Overload Bit for an IPv4 IS-IS Device.....	227
7.5.6 Checking the Configuration.....	228
7.6 Configuring IPv4 IS-IS Route Summarization.....	228
7.7 Configuring IPv4 IS-IS to Interact with Other Routing Protocols.....	229
7.7.1 Establishing the Configuration Task.....	229
7.7.2 Configuring a Preference Value for IPv4 IS-IS.....	231
7.7.3 Configuring IPv4 IS-IS to Advertise a Default Route.....	232

7.7.4 Configuring IPv4 IS-IS to Import External Routes.....	232
7.7.5 Checking the Configuration.....	234
7.8 Configuring the IPv4 IS-IS Route Convergence Speed.....	234
7.8.1 Establishing the Configuration Task.....	234
7.8.2 Configuring the Interval for Detecting IS-IS Neighboring Device Failures.....	235
7.8.3 Setting Flooding Parameters of SNPs and LSPs.....	236
7.8.4 Setting the SPF Calculation Interval.....	240
7.8.5 Configuring Convergence Priorities for IPv4 IS-IS Routes.....	241
7.8.6 Checking the Configuration.....	242
7.9 Configuring Static IPv4 BFD for IS-IS.....	243
7.10 Configuring Dynamic IPv4 BFD for IS-IS.....	245
7.11 Configuring Basic IPv6 IS-IS Functions.....	247
7.11.1 Establishing the Configuration Task.....	247
7.11.2 Creating IPv6 IS-IS Processes.....	248
7.11.3 Configuring IPv6 IS-IS Interfaces.....	250
7.11.4 (Optional) Configuring the IPv6 IS-IS Interfaces.....	252
7.11.5 (Optional) Configuring IPv6 IS-IS Attributes for Interfaces on Different Types of Networks.....	255
7.11.6 Checking the Configuration.....	257
7.12 Configuring IPv6 IS-IS Route Selection.....	258
7.12.1 Establishing the Configuration Task.....	258
7.12.2 Configuring IPv6 IS-IS Route Leaking.....	259
7.12.3 Configuring Principles for Using Equal-Cost IPv6 IS-IS Routes.....	260
7.12.4 Filtering IPv6 IS-IS Routes.....	261
7.12.5 Configuring an Overload Bit for an IPv6 IS-IS Device.....	261
7.12.6 Checking the Configuration.....	262
7.13 Configuring IPv6 IS-IS Route Summarization.....	262
7.14 Configuring IPv6 IS-IS to Interact with Other Routing Protocols.....	263
7.14.1 Establishing the Configuration Task.....	263
7.14.2 Configuring a Preference Value for IPv6 IS-IS.....	265
7.14.3 Configuring IPv6 IS-IS to Advertise a Default Route.....	266
7.14.4 Configuring IPv6 IS-IS to Import External Routes.....	266
7.14.5 Checking the Configuration.....	267
7.15 Configuring the IPv6 IS-IS Route Convergence Speed.....	268
7.15.1 Establishing the Configuration Task.....	268
7.15.2 Configuring the Interval for Detecting IS-IS Neighboring Device Failures.....	269
7.15.3 Setting Flooding Parameters of SNPs and LSPs.....	270
7.15.4 Setting the SPF Calculation Interval.....	274
7.15.5 Configuring Convergence Priorities for IPv6 IS-IS Routes.....	275
7.15.6 Checking the Configuration.....	275
7.16 Configuring IS-IS GR.....	276
7.16.1 Establishing the Configuration Task.....	276
7.16.2 Enabling IS-IS GR.....	277

7.16.3	Configuring Parameters of an IS-IS GR Session.....	277
7.16.4	Checking the Configuration.....	278
7.17	Maintaining IS-IS.....	278
7.17.1	Resetting IS-IS Data Structure.....	279
7.17.2	Resetting a Specific IS-IS Neighbor.....	279
7.18	Configuration Examples.....	280
7.18.1	Example for Configuring Basic IS-IS Functions.....	280
7.18.2	Example for Configuring the DIS Election of IS-IS.....	285
7.18.3	Example for Configuring Basic IS-IS IPv6 Functions.....	289
7.18.4	Example for Configuring IS-IS Fast Convergence.....	295
8	BGP Configuration.....	299
8.1	BGP Overview.....	301
8.2	BGP Features Supported by the AR150/200.....	301
8.3	Configuring Basic BGP Functions.....	308
8.3.1	Establishing the Configuration Task.....	308
8.3.2	Starting a BGP Process.....	309
8.3.3	Configuring BGP Peers.....	309
8.3.4	Configuring BGP to Import Routes.....	311
8.3.5	Checking the Configuration.....	313
8.4	Configuring BGP Route Attributes.....	314
8.4.1	Establishing the Configuration Task.....	314
8.4.2	Configuring the BGP Preference.....	315
8.4.3	Configuring Preferred Values for BGP Routes.....	316
8.4.4	Configuring a Default Local_Pref Attribute for a Device.....	316
8.4.5	Configuring MED Attributes for BGP Routes.....	317
8.4.6	Configuring Next_Hop Attributes for Routes.....	319
8.4.7	Configuring AS_Path Attributes for Routes.....	322
8.4.8	Checking the Configuration.....	325
8.5	Configuring BGP to Advertise Routes.....	326
8.5.1	Establishing the Configuration Task.....	326
8.5.2	Configuring BGP Filters.....	327
8.5.3	Configuring to Control the Advertisement of BGP Routing Information.....	333
8.5.4	Configuring BGP Soft Reset.....	334
8.5.5	Checking the Configuration.....	336
8.6	Configuring BGP to Receive Routes.....	337
8.6.1	Establishing the Configuration Task.....	337
8.6.2	Configuring BGP Filters.....	338
8.6.3	Configuring to Control the Acceptment of BGP Routing Information.....	344
8.6.4	Configuring BGP Soft Reset.....	346
8.6.5	Checking the Configuration.....	347
8.7	Configuring BGP Route Aggregation.....	348
8.8	Configuring BGP Peer Groups.....	350

8.8.1 Establishing the Configuration Task.....	350
8.8.2 Creating IBGP Peer Groups.....	351
8.8.3 Creating Pure EBGP Peer Groups.....	352
8.8.4 Creating Mixed EBGP Peer Groups.....	352
8.8.5 Checking the Configuration.....	353
8.9 Configuring BGP Route Reflectors.....	354
8.9.1 Establishing the Configuration Task.....	354
8.9.2 Configuring a Route Reflector and Specifying Clients.....	355
8.9.3 (Optional) Disabling Route Reflection Between Clients.....	356
8.9.4 (Optional) Configuring the Cluster ID for a Route Reflector.....	356
8.9.5 (Optional) Preventing BGP Routes from Being Added into the IP Routing Table.....	358
8.9.6 Checking the Configuration.....	359
8.10 Configuring a BGP Confederation.....	359
8.11 Configuring BGP Community Attributes.....	361
8.11.1 Establishing the Configuration Task.....	361
8.11.2 Configuring Community Attribute-Related Routing Policies.....	361
8.11.3 Configuring a BGP Device to Send Community Attributes to Its Peer.....	362
8.11.4 Checking the Configuration.....	363
8.12 Configuring Prefix-based BGP ORF.....	363
8.13 Configuring to Adjust the BGP Network Convergence Speed.....	365
8.13.1 Establishing the Configuration Task.....	365
8.13.2 Configuring a BGP ConnectRetry Timer.....	367
8.13.3 Configuring BGP Keepalive and Hold Timers.....	368
8.13.4 Configuring a MinRouteAdvertisementIntervalTimer.....	370
8.13.5 Disabling Fast Reset of EBGP Connections.....	371
8.13.6 Enabling BGP Tracking.....	372
8.13.7 Checking the Configuration.....	373
8.14 Configuring BGP Route Dampening.....	373
8.15 Configuring a BGP Device to Send a Default Route to Its Peer.....	374
8.16 Configuring BGP Load Balancing.....	376
8.17 Configuring Path MTU Auto Discovery.....	379
8.18 Configuring the BGP Next Hop Delayed Response.....	381
8.19 Configuring BFD for BGP.....	383
8.20 Configuring BGP GR.....	386
8.20.1 Establishing the Configuration Task.....	386
8.20.2 Enabling BGP GR.....	387
8.20.3 Configuring Parameters for a BGP GR Session.....	387
8.20.4 Checking the Configuration.....	388
8.21 Configuring BGP Security.....	388
8.21.1 Establishing the Configuration Task.....	388
8.21.2 Configuring MD5 Authentication.....	390
8.21.3 Configuring Keychain Authentication.....	390

8.21.4 Configuring BGP GTSM.....	391
8.21.5 Checking the Configuration.....	392
8.22 Maintaining BGP.....	393
8.22.1 Resetting BGP Connections.....	393
8.22.2 Clearing BGP Information.....	393
8.23 Configuration Examples.....	394
8.23.1 Example for Configuring Basic BGP Functions.....	394
8.23.2 Example for Configuring BGP Community Attributes for Routes.....	397
9 BGP4+ Configuration.....	401
9.1 BGP4+ Overview.....	403
9.2 BGP4+ Features Supported by the AR150/200.....	403
9.3 Configuring Basic BGP4+ Functions.....	403
9.3.1 Establishing the Configuration Task.....	404
9.3.2 Starting a BGP Process.....	404
9.3.3 Configuring an IPv6 Peer.....	405
9.3.4 (Optional) Configuring the Local Interfaces Used for BGP4+ Connections.....	407
9.3.5 Checking the Configuration.....	408
9.4 Configuring BGP4+ Route Attributes.....	408
9.4.1 Establishing the Configuration Task.....	408
9.4.2 Configuring the BGP4+ Preference.....	409
9.4.3 Configuring BGP4+ Preferred Value for Routing Information.....	410
9.4.4 Configuring the Default Local_Pref Attribute of the Local Router.....	410
9.4.5 Configuring the MED Attribute.....	411
9.4.6 Configuring the Next_Hop Attribute.....	412
9.4.7 Configuring the AS-Path Attribute.....	413
9.4.8 Configuring the BGP4+ Community Attribute.....	414
9.4.9 Checking the Configuration.....	415
9.5 Controlling the Advertising and Receiving of BGP4+ Routing Information.....	416
9.5.1 Establishing the Configuration Task.....	416
9.5.2 Configuring BGP4+ to Advertise Local IPv6 Routes.....	417
9.5.3 Configuring BGP4+ Route Aggregation.....	418
9.5.4 Configuring BGP4+ to Import and Filter External Routes.....	418
9.5.5 Configuring s to Advertise Default Routes to Peers.....	419
9.5.6 Configuring the Policy for Advertising BGP4+ Routing Information.....	420
9.5.7 Configuring the Policy for Receiving BGP4+ Routing Information.....	421
9.5.8 Configuring BGP4+ Soft Resetting.....	422
9.5.9 Checking the Configuration.....	423
9.6 Configuring Parameters of a Connection Between BGP4+ Peers.....	423
9.6.1 Establishing the Configuration Task.....	424
9.6.2 Configuring BGP4+ Timers.....	425
9.6.3 Configuring the Interval for Sending Update Packets.....	425
9.6.4 Setting the BGP4+ ConnectRetry Interval.....	426

9.6.5 Checking the Configuration.....	427
9.7 Configuring BGP4+ Tracking.....	427
9.7.1 Establishing the Configuration Task.....	428
9.7.2 Enabling BGP4+ Tracking.....	428
9.7.3 Checking the Configuration.....	429
9.8 Configuring BGP4+ Route Dampening.....	429
9.8.1 Establishing the Configuration Task.....	429
9.8.2 Enabling BGP4+ Route Dampening.....	430
9.8.3 Checking the Configuration.....	431
9.9 Configuring BGP4+ Load Balancing.....	431
9.10 Configuring a BGP4+ Peer Group.....	433
9.10.1 Establishing the Configuration Task.....	433
9.10.2 Creating an IBGP Peer Group.....	434
9.10.3 Creating a Pure EBGP Peer Group.....	435
9.10.4 Creating a Mixed EBGP Peer Group.....	436
9.10.5 Checking the Configuration.....	436
9.11 Configuring a BGP4+ Route Reflector.....	437
9.11.1 Establishing the Configuration Task.....	437
9.11.2 Configuring a Route Reflector and Specifying Clients.....	437
9.11.3 (Optional) Disabling a Route Reflection Between Clients.....	438
9.11.4 (Optional) Configuring the Cluster ID for a Route Reflector.....	439
9.11.5 Checking the Configuration.....	439
9.12 Configuring a BGP4+ Confederation.....	440
9.12.1 Establishing the Configuration Task.....	440
9.12.2 Configuring a BGP4+ Confederation Attribute.....	440
9.12.3 Checking the Configuration.....	441
9.13 Configuring BGP4+ Security.....	442
9.13.1 Establishing the Configuration Task.....	442
9.13.2 Configuring MD5 Authentication.....	443
9.13.3 Configuring Keychain Authentication.....	443
9.13.4 Configuring Basic BGP4+ GTSM Functions.....	444
9.13.5 Checking the Configuration.....	446
9.14 Maintaining BGP4+.....	446
9.14.1 Resetting BGP4+ Connections.....	446
9.14.2 Clearing BGP4+ Statistics.....	447
9.15 Configuration Examples.....	447
9.15.1 Example for Configuring Basic BGP4+ Functions.....	447
10 Routing Policy Configuration.....	452
10.1 Overview of the Routing Policy.....	454
10.2 Routing Policy Features Supported by the AR150/200.....	455
10.3 Configuring the IP-Prefix List.....	456
10.3.1 Establishing the Configuration Task.....	456

10.3.2 Configuring an IPv4 Prefix List.....	457
10.3.3 Configuring an IPv6 Prefix List.....	458
10.3.4 Checking the Configuration.....	459
10.4 Configuring the Route-Policy.....	459
10.4.1 Establishing the Configuration Task.....	459
10.4.2 Creating a Route-Policy.....	460
10.4.3 (Optional) Configuring the If-Match Clause.....	461
10.4.4 (Optional) Configuring the Apply Clause.....	462
10.4.5 Checking the Configuration.....	463
10.5 Applying Filters to Received Routes.....	463
10.5.1 Establishing the Configuration Task.....	464
10.5.2 Filtering Routes Received by RIP.....	465
10.5.3 Filtering Routes Received by OSPF.....	465
10.5.4 Filtering Routes Received by IS-IS.....	466
10.5.5 Filtering Routes Received by BGP.....	466
10.5.6 Checking the Configuration.....	467
10.6 Applying Filters to Advertised Routes.....	468
10.6.1 Establishing the Configuration Task.....	468
10.6.2 Filtering Routes Advertised by RIP.....	469
10.6.3 Filtering Routes Advertised by OSPF.....	469
10.6.4 Filtering Routes Advertised by IS-IS.....	470
10.6.5 Filtering Routes Advertised by BGP.....	470
10.6.6 Checking the Configuration.....	472
10.7 Applying Filters to Imported Routes.....	472
10.7.1 Establishing the Configuration Task.....	472
10.7.2 Applying Route-Policy to Routes Imported by RIP.....	473
10.7.3 Applying Route-Policy to Routes Imported by OSPF.....	474
10.7.4 Applying Route-Policy to Routes Imported by IS-IS.....	474
10.7.5 Applying Route-Policy to Routes Imported by BGP.....	475
10.7.6 Checking the Configuration.....	475
10.8 Controlling the Valid Time of the Routing policy.....	476
10.8.1 Establishing the Configuration Task.....	476
10.8.2 Configuring the Delay for Applying the Routing Policy.....	477
10.8.3 Checking the Configuration.....	477
10.9 Maintaining the Routing Policy.....	478
10.10 Configuration Examples.....	478
10.10.1 Example for Filtering Received and Advertised Routes.....	478

1 IP Routing Basic Configuration

About This Chapter

This chapter describes IP routing, which functions as the basis for data communication networks.

[1.1 Routing Management](#)

To forward data, routers need to establish and refresh routing tables and forward packets according to the information in routing tables.

1.1 Routing Management

To forward data, routers need to establish and refresh routing tables and forward packets according to the information in routing tables.

1.1.1 Displaying of the Routing Table

Routing tables are one of the best sources of information about a network. Checking these tables and helps you to locate faults.

Prerequisites

Checking information about routing tables helps fault location. The following lists common commands for displaying information about the routing table.

Note that display commands can be run in all views.

Procedure

- Run the **display ip routing-table** command to check brief information about current active routes.
- Run the **display ip routing-table verbose** command to check detailed information about the IP routing table.
- Run the **display ip routing-table ip-address** [*mask* | *mask-length*] [**longer-match**] [**verbose**] command to check the routes to a specified destination address.
- Run the **display ip routing-table ip-address1** { *mask1* | *mask-length1* } *ip-address2* { *mask2* | *mask-length2* } [**verbose**] command to check the routes whose destination addresses are within a specified address range.
- Run the **display ip routing-table acl** { *acl-number* | *acl-name* } [**verbose**] command to check the routes filtered by a specified basic ACL.
- Run the **display ip routing-table ip-prefix ip-prefix-name** [**verbose**] command to check the routes filtered by a specified IP prefix list.
- Run the **display ip routing-table statistics** command to check statistics about the IP routing table.
- Run the **display ip routing-table vpn-instance vpn-instance-name** command to check brief information about the VPN routing table.
- Run the **display ip routing-table vpn-instance vpn-instance-name verbose** command to check detailed information about the VPN routing table.

----End

1.1.2 Displaying of the Routing Management Module

By using the display commands to view information about the routing management (RM) module, you can locate routing problems.

Context

You can use display commands of the RM module to locate routing problems.

The **display** commands can be run in all views.

Procedure

- Run the **display rm interface** [*interface-type interface-number*] command to check RM information about a specified interface.
- Run the **display rm interface vpn-instance** *vpn-instance-name* command to check RM information about the private network interface.

----End

2 IP Static Route Configuration

About This Chapter

Static routes are commonly used on simple networks. Properly configuring and using static routes can improve network performance and help ensure enough bandwidth is available for important services.

[2.1 Static Route](#)

Static routes are special routes that network administrators must manually configure.

[2.2 Static Routing Features Supported by the AR150/200](#)

The system supports various static route features, including IPv4 static routes, default routes, BFD for static routes, and permanent advertisement of static routes.

[2.3 Configuring an IPv4 Static Route](#)

On an IPv4 network, you can accurately control route selection by configuring IPv4 static routes.

[2.4 Configuring an IPv6 Static Route](#)

On an IPv6 network, you can accurately control route selection by configuring IPv6 static routes.

[2.5 Configuring BFD for IPv4 Static Routes on the Public Network](#)

On an IPv4 network, configuring BFD for IPv4 static routes on the public network can speed up route convergence and improve network reliability.

[2.6 Configuring NQA for IPv4 Static Routes](#)

On an IPv4 network, if Bidirectional Forwarding Detection (BFD) for static IPv4 routes on the public network cannot be configured because one of the communicating devices does not support BFD, Network Quality Analysis (NQA) for static IPv4 routes can be configured to detect faults in links. An NQA test instance is used to detect the link status to allow a fast link switchover after a fault occurs in a link. This switchover helps prevent prolonged service interruptions.

[2.7 Configuration Examples](#)

Static route configuration examples explain networking requirements, networking diagrams, configuration notes, configuration roadmap, and configuration procedures.

2.1 Static Route

Static routes are special routes that network administrators must manually configure.

On a simple network topology, you only need to configure static routes so that the network can run properly. Properly using static routes improves the network performance and provides the guaranteed bandwidth for important applications.

The disadvantage of static routes is that if a fault occurs on the network or the network topology changes, static routes cannot automatically change and must be changed manually by the administrator.

2.2 Static Routing Features Supported by the AR150/200

The system supports various static route features, including IPv4 static routes, default routes, BFD for static routes, and permanent advertisement of static routes.

IPv4 Static Route

IPv4 static routes need to be manually configured by the administrator. IPv4 static routes are applicable to simple IPv4 networks.

An IPv4 static route is an IPv4 default route if its destination address is 0.0.0.0 and the mask length is 0.

If the destination address of an IPv4 packet fails to match any entry in the routing table, the router uses the IPv4 default route to forward the IPv4 packet.

The AR150/200 supports ordinary static routes. For details of VPN instances, see the *Huawei AR150&200 Series Enterprise Routers Feature Description - VPN*.

IPv6 Static Route

Similar to IPv4 static routes, IPv6 static routes need to be manually configured by the administrator. IPv6 static routes are applicable to simple IPv6 networks.

If the destination address of an IPv6 static route is `::/0` with the mask length being 0, this IPv6 static route is an IPv6 default route.

If the destination address of an IPv6 packet fails to match any entry in the routing table, the router uses the IPv6 default route to forward the IPv6 packet.

NOTE

The main differences between IPv6 static routes and IPv4 static routes are their destination addresses and next-hop addresses. The next-hop address of an IPv6 static route is an IPv6 address, whereas the next-hop address of an IPv4 static route is an IPv4 address.

The IPv6 static routing function is used with a license. To use the IPv6 static routing function, apply for and purchase the following license from the Huawei local office:

- AR150&200 Value-Added Data Package

Default Route

Default routes are a special type of routes that are usually configured by network administrators. Default routes can also be generated by dynamic routing protocols such as Open Shortest Path First (OSPF) or Intermediate System-to-Intermediate System (IS-IS).

Default routes are used only when packets to be forwarded fail to match any entry in the routing table. You can run the **display ip routing-table** command to check whether the default route is configured.

If the destination address of a packet does not match any entry in the routing table, the router uses the default route to forward the packet. If no default route exists, the packet is discarded, and an Internet Control Message Protocol (ICMP) packet is sent to inform the originating host that the destination host or network is unreachable.

BFD for Static Route

Unlike dynamic routing, static routing does not have a detection mechanism. If a fault occurs on the network, administrator involvement is required. Bidirectional Forwarding Detection (BFD) for static route is used to bind BFD sessions to static routes on the public network. The BFD sessions are used to detect the link status of a static route. The system then uses the detection results to determine whether to add static routes to its IP routing table.

After BFD for static route is configured, each static route can be bound to a BFD session.

- When the BFD session on the link of a static route detects that the link changes from Up to Down, BFD reports the fault to the RM module, and then the RM module sets the route to inactive. Subsequently, the route becomes unavailable and is deleted from the routing table.
- When a BFD session is established on the link of a static route (the link changes from Down to Up), BFD reports the success to the RM module, and then the RM module sets the route to active. Subsequently, the route becomes available and is added to the IP routing table.

Permanent Advertisement of Static Routes

Permanent advertisement of static routes provides way to monitor services and a simple link detection mechanism which also improves compatibility between Huawei devices and non-Huawei devices. If service traffic needs to be forwarded along a specified path, you can detect links by pinging the destination addresses of static routes.

2.3 Configuring an IPv4 Static Route

On an IPv4 network, you can accurately control route selection by configuring IPv4 static routes.

2.3.1 Establishing the Configuration Task

Before configuring an IPv4 static route, familiarize yourself with the applicable environment, complete the pre-configuration tasks, and obtain the data required for the configuration. This will help you complete the configuration task quickly and accurately.

Applicable Environment

When configuring an IPv4 static route, note the following:

- Destination address and mask
 In the **ip route-static** command, the IPv4 destination address is in dotted decimal notation, and the mask can be either expressed in dotted decimal notation or replaced by the mask length (namely, the number of consecutive 1s in the mask).
- Outbound interface and next-hop address
 When configuring a static route, you can specify either *interface-type interface-number* or *nexthop-address* depending on which parameter is better suited to your situation.
 In real-world situations, each routing entry requires a next-hop address. When sending a packet, the router first searches for the matched route in the routing table against the destination address.
 For example, in some cases, the link layer is encapsulated with PPP, you can also specify outbound interfaces when configuring the router even if the remote address is not known. In this manner, it is unnecessary to modify the router configuration if the remote address changes.
- Other attributes
 Setting different preferences for static routes helps flexibly apply routing policies. For example, when configuring multiple routes to the same destination address, you can set the same preference for these routes to implement load balancing. You can also set different preferences to implement routing redundancy.
 When the **ip route-static** command is run to configure a static route, a default route is configured if the destination address and the mask are set to all 0s (0.0.0.0 0.0.0.0).

Pre-configuration Tasks

Before configuring an IPv4 static route, complete the following task:

- Configuring link layer protocol parameters and IP addresses for interfaces to ensure that the link layer protocol status of the interfaces is Up

Data Preparation

To configure an IPv4 static route, you need the following data.

No.	Data
1	Destination address and mask
2	Outbound interface or next-hop IPv4 address
3	Preference of the IPv4 static route

2.3.2 Configuring an IPv4 Static Route on the Public Network

When configuring an IPv4 static route, configure its destination address, outbound interface, and next hop.

Context

Do as follows on the router to be configured with a static route:

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
ip route-static ip-address { mask | mask-length } { nexthop-address | interface-  
type interface-number [ nexthop-address ] | vpn-instance vpn-instance-name nexthop-  
address } [ preference preference | tag tag ] * [ description text ]
```

An IPv4 static route is configured.

By default, no IPv4 static route is configured.

----End

2.3.3 (Optional) Setting the Default Preference for IPv4 Static Routes

Setting the default preference for IPv4 static routes can affect route selection.

Context

Do as follows on the routers that need to be configured with static routes and change the default preference for static routes:

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
ip route-static default-preference preference
```

The default preference is set for static routes.

By default, the preference of static routes is 60.

When a static route is configured, the default preference is used if no preference is explicitly specified for the static route. After a default preference is specified, the new default preference is valid for subsequent rather than existing IPv4 static routes.

----End

2.3.4 (Optional) Configuring Static Route Selection Based on Relay Depth

After static route selection based on relay depths is configured, the static route module selects the static route with the smallest relay depth as the active route and delivers it to the FIB table. The other routes become inactive.

Context

After static routes are configured, multiple static routes with the same prefix and preference but different relay depths exist. After static route selection based on relay depths is configured, the static route module selects the route with the smallest relay depth as the active route and delivers it to the Forwarding Information Base (FIB) table. The other routes become inactive.

Do as follows on the router to be configured with static routes:

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
ip route-static selection-rule relay-depth
```

Static route selection based on relay depths is configured.

By default, static routes are not selected according to relay depths.

----End

2.3.5 (Optional) Configuring Permanent Advertisement of IPv4 Static Routes

Permanent advertisement of static routes provides a low-cost and simple link detection mechanism and improves the compatibility between Huawei devices and non-Huawei devices.

Context

Link connectivity directly affects the stability and availability of a network. Consequently, detecting link status is an important part of network maintenance. If service traffic needs to be forwarded along a specified path, you can detect the status of the path using a ping operation. In this manner, you can monitor services at a very low cost.

With permanent advertisement of static routes, you can detect link connectivity by pinging the destination addresses of static routes. After permanent advertisement of static routes is configured, static routes always take effect regardless of the outbound interface status. In this case, the system forwards ping packets along a specified path only, which helps detect the link status of the specified path.

Do as follows on the router where IPv4 static routes need to be configured.

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
ip route-static ip-address { mask | mask-length } { nexthop-address | interface-  
type interface-number [ nexthop-address ] | vpn-instance vpn-instance-name nexthop-  
address } permanent
```

Permanent advertisement of IPv4 static routes is configured.

By default, permanent advertisement of IPv4 static routes is not configured.

----End

2.3.6 Checking the Configuration

After an IPv4 static route is configured, you can check detailed information about the configured IPv4 static route.

Prerequisites

The configurations for an IPv4 static route are complete.

Procedure

- Run the **display ip routing-table** command to check brief information about the IPv4 routing table.
- Run the **display ip routing-table verbose** command to check detailed information about the IPv4 routing table.

----End

2.4 Configuring an IPv6 Static Route

On an IPv6 network, you can accurately control route selection by configuring IPv6 static routes.

2.4.1 Establishing the Configuration Task

Before configuring an IPv6 static route, familiarize yourself with the applicable environment, complete the pre-configuration tasks, and obtain the data required for the configuration. This will help you complete the configuration task quickly and accurately.

Applicable Environment

On a small IPv6 network, you can implement network interconnection by configuring IPv6 static routes. Compared with using dynamic routes, using static routes saves the bandwidth.

Pre-configuration Tasks

Before configuring an IPv6 static route, complete the following task:

- Configuring link layer protocol parameters and IP addresses for interfaces to ensure that the link layer protocol status of the interfaces is Up

Data Preparation

To configure an IPv6 static route, you need the following data.

No.	Data
1	Destination address and mask

No.	Data
2	Outbound interface or next-hop IPv6 address
3	Preference of the IPv6 static route

2.4.2 Configuring an IPv6 Static Route on the Public Network

When configuring an IPv6 static route, you need to correctly configure its destination address, outbound interface, and next hop.

Context

Do as follows on the router to be configured with static routes:

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
ipv6 route-static dest-ipv6-address prefix-length { interface-type interface-number [ nexthop-  
ipv6-address ] | nexthop-ipv6-address } [ preference preference | tag tag ] * [ description  
text ]
```

An IPv6 static route is configured.

When configuring a static route, you need to specify either the outbound interface or the next-hop address according to the actual situation. If the outbound interface is a non-P2P interface, you must also specify the next-hop address in addition to specifying the outbound interface.

If **preference** is not specified, the default preference is 60.

By default, no IPv6 static route is configured.

---End

2.4.3 (Optional) Setting the Default Preference for IPv6 Static Routes

By setting the default preference for an IPv6 static route, you can change the preference of the static route.

Context

Do as follows on the router that need to be configured with IPv6 static routes and change the default priority for IPv6 static routes:

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
ipv6 route-static default-preference preference
```

The default preference of IPv6 static routes is set.

By default, the preference of IPv6 static routes is 60.

When an IPv6 static route is configured, the default preference is used if the preference of the static route is not explicitly specified. After the default preference is specified, the default preference is valid for subsequent rather than existing IPv6 static routes.

----End

2.4.4 Checking the Configuration

After an IPv6 static route is configured, you can check detailed information about the configured route.

Prerequisites

The configurations of an IPv6 static route are complete.

Procedure

- Run the **display ipv6 routing-table** command to check brief information about the IPv6 routing table.
- Run the **display ipv6 routing-table verbose** command to check detailed information about the IPv6 routing table.

----End

2.5 Configuring BFD for IPv4 Static Routes on the Public Network

On an IPv4 network, configuring BFD for IPv4 static routes on the public network can speed up route convergence and improve network reliability.

2.5.1 Establishing the Configuration Task

Before configuring BFD for static routes, familiarize yourself with the applicable environment, complete the pre-configuration tasks, and obtain the data required for the configuration. This will help you complete the configuration task quickly and accurately.

Applicable Environment

BFD can quickly detect IPv4 forwarding failures, ensuring QoS for voice, video, and other video-on-demand (VoD) services on an IPv4 network. With BFD, service providers can provide voice over IP (VoIP) and other real-time services with high availability and scalability.

By binding IPv4 static routes to BFD sessions, you can use BFD sessions to provide link detection for IPv4 static routes on the public network. A static route can be bound to a BFD session.

 **NOTE**

A BFD session currently does not detect route switching. If the change of bound peer IP address causes a route to switch to another link, the BFD session is negotiated again only when the original link fails.

Pre-configuration Tasks

Before configuring BFD for IPv4 static routes on the public network, complete the following task:

- Configuring link layer protocol parameters and IP addresses for interfaces to ensure that the link layer protocol status of the interfaces is Up

Data Preparation

To configure BFD for IPv4 static routes on the public network, you need the following data.

No.	Data
1	Destination address and mask
2	Outbound interface or next-hop IPv4 address
3	IP address of the peer detected by BFD
4	Local discriminator and remote discriminator of a BFD session

2.5.2 Configuring an IPv4 Static Route on the Public Network

When configuring an IPv4 static route, configure its destination address, outbound interface, and next hop.

Context

Do as follows on the router to be configured with a static route:

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
ip route-static ip-address { mask | mask-length } { nexthop-address | interface-  
type interface-number [ nexthop-address ] | vpn-instance vpn-instance-name nexthop-  
address } [ preference preference | tag tag ] * [ description text ]
```

An IPv4 static route is configured.

By default, no IPv4 static route is configured.

----End

2.5.3 Configuring a BFD Session

BFD sessions are used to quickly detect and monitor the connectivity of links on a network.

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
bfd
```

BFD is enabled globally and the BFD view is displayed.

Step 3 Run:

```
quit
```

Return to the system view.

Step 4 Run the `bfd cfg-name bind peer-ip peer-ip [vpn-instance vpn-instance-name] [interface interface-type interface-number] [source-ip source-ip]` command to configure a BFD session.

- When a BFD session is set up for the first time, you need to bind the peer IP address to it. After the BFD session is set up, you cannot modify it.
- When the BFD configuration items are created, the system checks only the format of the IP address rather than the correctness. The BFD session cannot be established if incorrect peer IP address or source IP address is bound.
- When the IP address of the peer and the local interface are both specified, a single-hop link is monitored. BFD monitors the route with the outbound interface specified and *peer-ip* as the next-hop IP address specified. When only the IP address of the peer is specified, multi-hop routes are monitored.
- When the BFD and URPF are used together, URPF checks the source IP address of the received packets. Therefore, when creating a BFD binding, you need to specify the source IP address of the BFD packet in case the BFD packet is incorrectly discarded.

Step 5 Configure the discriminators.

- Run:

```
discriminator local discr-value
```

The local discriminator is configured.

- Run:

```
discriminator remote discr-value
```

The remote discriminator is configured.

 **NOTE**

The local discriminator of the local device corresponds to the remote discriminator of the remote device, and the remote discriminator of the local device corresponds to the local discriminator of the remote device. The local discriminator of the local device must be the same as the remote discriminator of the remote device. Otherwise, the session cannot be correctly set up. After the local and remote discriminators are configured, they cannot be modified.

Step 6 Run:

```
commit
```

The configurations are committed.

 **NOTE**

When setting up a BFD session, you must run the **commit** command after configuring necessary parameters, such as local and remote discriminators; otherwise, the session cannot be set up.

----End

2.5.4 Binding a Static Route to a BFD Session

When binding a static route to a BFD session, ensure that the static route resides on the same link as the BFD session.

Context

Do as follows on the router to bind a static route to a BFD session:

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
ip route-static ip-address { mask | mask-length } { nexthop-address | interface-  
type interface-number [ nexthop-address ] } [ preference preference | tag tag ] *  
track bfd-session cfg-name [ description text ]
```

A BFD session is bound to the IPv4 static route on the public network.

 **NOTE**

When binding a static route to a BFD session, ensure that the static route resides on the same link as the BFD session.

----End

2.5.5 Checking the Configuration

After BFD for static route is configured, you can check information about BFD sessions and BFD for static route.

Prerequisites

BFD configurations for IPv4 static routes are complete.

Procedure

- Run the **display bfd session** { **all** | **discriminator** *discr-value* } [**verbose**] command to check BFD session information.
- Run the **display current-configuration | include bfd** command to check the configuration of BFD for static routes.

You can check information about a BFD session only after parameters for the BFD session are set and the BFD session is established.

If BFD session negotiation succeeds, the status of the BFD session is displayed as **Up**. You can also check that the BFD session is bound to the static route by running the **display current-configuration | include bfd** command in the system view.

---End

2.6 Configuring NQA for IPv4 Static Routes

On an IPv4 network, if Bidirectional Forwarding Detection (BFD) for static IPv4 routes on the public network cannot be configured because one of the communicating devices does not support BFD, Network Quality Analysis (NQA) for static IPv4 routes can be configured to detect faults in links. An NQA test instance is used to detect the link status to allow a fast link switchover after a fault occurs in a link. This switchover helps prevent prolonged service interruptions.

2.6.1 Establishing the Configuration Task

Before configuring NQA for static IPv4 routes, familiarize yourself with the applicable environment, complete the pre-configuration tasks, and obtain the required data. This will help you complete the configuration task quickly and accurately.

Applicable Environment

In real world situations, the link status is monitored for network stability. If an active link fails, traffic switches to a standby link to ensure non-stop traffic forwarding. The Address Resolution Protocol (ARP) probe function and BFD are usually used to detect link faults. In addition, Interior Gateway Protocol (IGP) convergence helps reveal link faults. In certain situations, the preceding methods are unsuitable. For example:

- If only one link, not every link, on the network needs to be monitored, the ARP detection is unsuitable.
- If any device on the network does not support BFD, BFD is unavailable.
- If either end of a link is a Layer 2 device, dynamic routing protocols cannot be deployed. As a result, no IGP convergence occurs.

In these situations, NQA for static IPv4 routes can be configured to detect link faults. It can be used to detect faults in links where Layer 2 devices reside and take effect even if only one of the two communicating devices supports NQA.

If a fault occurs, an NQA test instance can immediately detect the fault and instruct the system to delete the associated static route from the IP routing table. Traffic is then forwarded along another path.

Pre-configuration Tasks

Before configuring NQA for static IPv4 routes, complete the following task:

- Configuring link layer protocol parameters and IP addresses for interfaces to ensure that the link layer protocol on the interfaces is Up

Data Preparation

To configure NQA for static IPv4 routes, you need the following data.

No.	Data
1	Administrator name and name of an NQA test instance
2	Destination IP address of the NQA test instance
3	Destination network address and mask

2.6.2 Configuring an ICMP Type NQA Test Instance

NQA is an effective tool for locating and diagnosing network faults.

Context

NQA measures the performance of different protocols running on a network. With NQA, carriers can collect the operation indicators of networks in real time. These indicators include total delay of the Hypertext Transfer Protocol (HTTP), delay in the Transfer Control Protocol (TCP) connection, delay in Domain Name Server (DNS) resolution, file transmission rate, delay in the File Transfer Protocol (FTP) connection, and DNS resolution error ratio. To check these performance indexes, you can create NQA test instances.

An NQA test is performed between a client and a server. The client is responsible for initiating an NQA test. After test instances are configured on the client, NQA places these test instances into test instance queues according to their operation types. After the test instances are started, data information about the protocol-related running status can be collected based on information about the return packets.

An Internet Control Messages Protocol (ICMP) NQA test instance checks whether a route from the NQA client to the destination is reachable. The ICMP NQA test performs a function similar to the **ping** command but provides more detailed output:

- By default, the output contains the five most recent tests..
- The test result contains information including the average delay, packet loss ratio, and time when the last packet was correctly received.

The minimum interval at which an ICMP NQA test instance sends packets is once per second, which ensures that NQA reports the test results to the system when a link fault is detected and when the link fault is rectified. For details about NQA, see the chapter "NQA Configuration" in the *Huawei AR150&200 Series Enterprise Routers Configuration Guide - Network Management*.

Do as follows on the NQA client:

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
nqa test-instance admin-name test-name
```

An NQA test instance is created and the test instance view is displayed.

Step 3 Run:

```
test-type icmp
```

The test type is set to ICMP.

Step 4 Run:

```
destination-address ipv4 ip-address
```

The destination address is specified for the NQA test instance.

Step 5 (Optional) Run:

```
frequency interval
```

The interval for automatically performing an NQA test is set. By default, no interval is set, and only one test is performed.

Step 6 (Optional) Run:

```
probe-count number
```

The number of probes to be sent each time is set for the NQA test instance. By default, three probes are sent each time.

After probes are sent multiple times for the NQA test instance, you can estimate the network quality more accurately based on the collected statistics.

Step 7 Run:

```
start
```

The NQA test instance is started.

Run one of the following commands in different situations:

- To start an NQA test immediately, run the **start now [end { at [yyyy/mm/dd] hh:mm:ss | delay { seconds second | hh:mm:ss } | lifetime { seconds second | hh:mm:ss } }]** command.
- To start an NQA test at a specified time, run the **start at [yyyy/mm/dd] hh:mm:ss [end { at [yyyy/mm/dd] hh:mm:ss | delay { seconds second | hh:mm:ss } | lifetime { seconds second | hh:mm:ss } }]** command.
- To start an NQA test after a certain period of time, run the **start delay { seconds second | hh:mm:ss } [end { at [yyyy/mm/dd] hh:mm:ss | delay { seconds second | hh:mm:ss } | lifetime { seconds second | hh:mm:ss } }]** command.

---End

2.6.3 Binding an IPv4 Static Route to an NQA Test Instance

If a static IPv4 route is associated with an NQA test instance, NQA tests the link status periodically. After NQA detects a fault in the link related to the associated static route, the static route is deleted and traffic diverts to another path.

Context

On a network with a simple topology, configuring static routes is usually adequate enough to ensure the network is able to operate correctly. Static routes can also be configured on a router that cannot run dynamic routing protocols to generate routes to the destination. Unlike dynamic routing protocols, static routes do not have a dedicated detection mechanism. Static routes cannot detect faults in the network, which means that traffic loss will likely occur at some point.

The NQA for static IPv4 routes feature allows static IPv4 routes to be associated with NQA test instances. The ping function of NQA test instances is used to check the status of links through which static routes pass. If a fault occurs in the link for a static route, the system deletes the static route to force traffic transmitted based on this route to divert to another path.

Do as follows on the router that requires NQA for static IPv4 routes:

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
ip route static ip-address { mask | mask-length } { nexthop-address | interface-  
type interface-number [ nexthop-address ] } [ preference preference | tag tag ] *  
track nqa admin-name test-name [ description text ]
```

A static IPv4 route is associated with an NQA test instance.

NOTE

The destination address of an NQA test instance cannot be the destination address of an associated static route.

If the static route associated with one NQA test instance is associated with another NQA test instance, the association between the static route and the former NQA test instance is automatically removed.

---End

2.6.4 Checking the Configuration

After associating an NQA test instance with a static route, you can check NQA test results and information about the association between the static route and the NQA test instance.

Prerequisites

NQA configurations for static IPv4 routes are complete.

NOTE

NQA test results cannot be displayed automatically on the terminal. To check NQA test results, run the **display nqa results** command. By default, the command output shows the results of the five most recent tests.

Procedure

Step 1 Run the **display current-configuration | include nqa** command to check NQA configurations for static IPv4 routes.

Step 2 Run the **display nqa results** [**test-instance admin-name test-name**] command to check NQA test results.

----End

Example

After associating a static route to an NQA test instance, run the **display current-configuration | include nqa** command in the system view to check whether the static route has been associated with the NQA test instance. For example:

```
<Huawei> display current-configuration | include nqa
ip route-static 172.16.1.3 255.255.255.255 Ethernet1/0/0 track nqa admin icmp
nqa test-instance admin icmp
```

Run the **display nqa results** command. The test records are successfully queried if the following information is displayed:

- testflag is active
- testtype is icmp
- The test is finished
- Completion:success

For example:

```
<Huawei> display nqa results test-instance admin icmp
NQA entry(admin, icmp) :testflag is active ,testtype is icmp
 1 . Test 206 result   The test is finished
  Send operation times: 15          Receive response times: 15
  Completion:success          RTD OverThresholds number: 0
  Attempts number:1          Drop operation number:0
  Disconnect operation number:0  Operation timeout number:0
  System busy operation number:0  Connection fail number:0
  Operation sequence errors number:0  RTT Stats errors number:0
  Destination ip address:172.16.1.2
  Min/Max/Average Completion Time: 30/50/35
  Sum/Square-Sum Completion Time: 530/19900
  Last Good Probe Time: 2010-10-25 15:39:57.1
  Lost packet ratio: 0 %
```

For an ICMP NQA test, the minimum, maximum, and average time for receiving ICMP Echo-Reply packets are displayed as **Min/Max/Average Completion Time**. In addition, the NQA test packet loss ratio is displayed, which helps provide a clear indication of the link status. In the preceding example, the packet loss ratio is 0, indicating that the link works properly.

NOTE

If the **frequency interval** command is configured for an NQA test instance, **testflag is active** is displayed. If the **frequency interval** command is not configured for an NQA test instance, the NQA test is performed only once and **testflag is inactive** is displayed.

2.7 Configuration Examples

Static route configuration examples explain networking requirements, networking diagrams, configuration notes, configuration roadmap, and configuration procedures.

2.7.1 Example for Configuring IPv4 Static Routes

You can configure IPv4 static routes to interconnect any two devices on an IPv4 network.

Networking Requirements

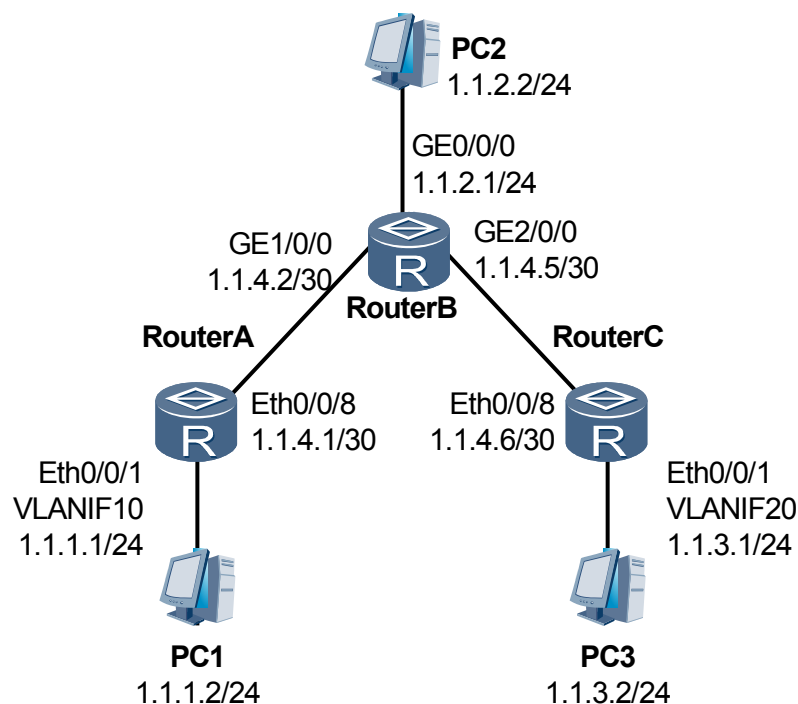
Figure 2-1 shows IP addresses and masks of interfaces and hosts. Any two hosts in **Figure 2-1** are required to be interconnected by using static routes.



NOTE

AR150/200 is RouterA, or RouterC.

Figure 2-1 Networking diagram for configuring IPv4 static routes



Configuration Roadmap

The configuration roadmap is as follows:

1. Configure an IPv4 address for each interface on each router for interworking.
2. Configure an IPv4 static route and a default route to the destination address on each router.
3. Configure an IPv4 default gateway on each host to make any two hosts communicate.

Data Preparation

To complete the configuration, you need the following data:

- Default route with the next hop being 1.1.4.2 of Router A
- Static route with the destination address and next hop being 1.1.1.0 and 1.1.4.1 respectively of Router B
- Static route with the destination address and next hop being 1.1.3.0, and 1.1.4.6 respectively of Router B

- Default route with the next hop being 1.1.4.5 of Router C
- Default gateway addresses of PC1, PC2, and PC3 being 1.1.1.1, 1.1.2.1, and 1.1.3.1 respectively

Procedure

Step 1 Configure an IP address for each interface.

The configuration details are not described here.

Step 2 Configure static routes.

Configure an IPv4 default route on Router A.

```
[RouterA] ip route-static 0.0.0.0 0.0.0.0 1.1.4.2
```

Configure two IPv4 static routes on Router B.

```
[RouterB] ip route-static 1.1.1.0 255.255.255.0 1.1.4.1  
[RouterB] ip route-static 1.1.3.0 255.255.255.0 1.1.4.6
```

Configure an IPv4 default route on Router C.

```
[RouterC] ip route-static 0.0.0.0 0.0.0.0 1.1.4.5
```

Step 3 Configure hosts.

Set default gateway addresses of PC1, PC2, and PC3 to 1.1.1.1, 1.1.2.1, and 1.1.3.1 respectively.

Step 4 Verify the configuration.

Check the IP routing table of Router A.

```
[RouterA] display ip routing-table  
Route Flags: R - relay, D - download to fib  
-----  
Routing Tables: Public  
Destinations : 8 Routes : 8  
Destination/Mask Proto Pre Cost Flags NextHop Interface  
0.0.0.0/0 Static 60 0 RD 1.1.4.2 Ethernet0/0/8  
1.1.1.0/24 Direct 0 0 D 1.1.1.1 Vlanif10  
1.1.1.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0  
1.1.4.0/30 Direct 0 0 D 1.1.4.1 Ethernet0/0/8  
1.1.4.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0  
1.1.4.2/32 Direct 0 0 D 1.1.4.2 Ethernet0/0/8  
127.0.0.0/8 Direct 0 0 D 127.0.0.1 InLoopBack0  
127.0.0.1/32 Direct 0 0 D 127.0.0.1 InLoopBack0
```

Run the **ping** command to verify the connectivity.

```
[RouterA] ping 1.1.3.1  
PING 1.1.3.1: 56 data bytes, press CTRL_C to break  
Reply from 1.1.3.1: bytes=56 Sequence=1 ttl=254 time=62 ms  
Reply from 1.1.3.1: bytes=56 Sequence=2 ttl=254 time=63 ms  
Reply from 1.1.3.1: bytes=56 Sequence=3 ttl=254 time=63 ms  
Reply from 1.1.3.1: bytes=56 Sequence=4 ttl=254 time=62 ms  
Reply from 1.1.3.1: bytes=56 Sequence=5 ttl=254 time=62 ms  
--- 1.1.3.1 ping statistics ---  
5 packet(s) transmitted  
5 packet(s) received  
0.00% packet loss  
round-trip min/avg/max = 62/62/63 ms
```

Run the **tracert** command to verify the connectivity.

```
[RouterA] tracert 1.1.3.1
```

```
tracert to 1.1.3.1(1.1.3.1), max hops: 30 ,packet length: 40,press CTRL_C to
break
 1 1.1.4.2 31 ms  32 ms  31 ms
 2 1.1.4.6 62 ms  63 ms  62 ms
```

----End

Configuration Files

- Configuration file of Router A

```
#
 sysname RouterA
#
vlan batch 10
#
interface Vlanif10
 ip address 1.1.1.1 255.255.255.0
#
interface Ethernet0/0/1
 port link-type access
 port default vlan 10
#
interface Ethernet0/0/8
 ip address 1.1.4.1 255.255.255.252
#
ip route-static 0.0.0.0 0.0.0.0 1.1.4.2
#
return
```

- Configuration file of Router B

```
#
 sysname RouterB
#
interface GigabitEthernet0/0/0
 ip address 1.1.2.1 255.255.255.0
#
interface GigabitEthernet1/0/0
 ip address 1.1.4.2 255.255.255.252
#
interface GigabitEthernet2/0/0
 ip address 1.1.4.5 255.255.255.252
#
ip route-static 1.1.1.0 255.255.255.0 1.1.4.1
ip route-static 1.1.3.0 255.255.255.0 1.1.4.6
#
return
```

- Configuration file of Router C

```
#
 sysname RouterC
#
vlan batch 20
#
interface Vlanif20
 ip address 1.1.3.1 255.255.255.0
#
interface Ethernet0/0/1
 port link-type access
 port default vlan 20
#
interface Ethernet0/0/8
 ip address 1.1.4.6 255.255.255.252
#
ip route-static 0.0.0.0 0.0.0.0 1.1.4.5
#
return
```


2.7.2 Example for Configuring IPv6 Static Routes

You can configure IPv6 static routes to interconnect any two devices on an IPv6 network.

Networking Requirements

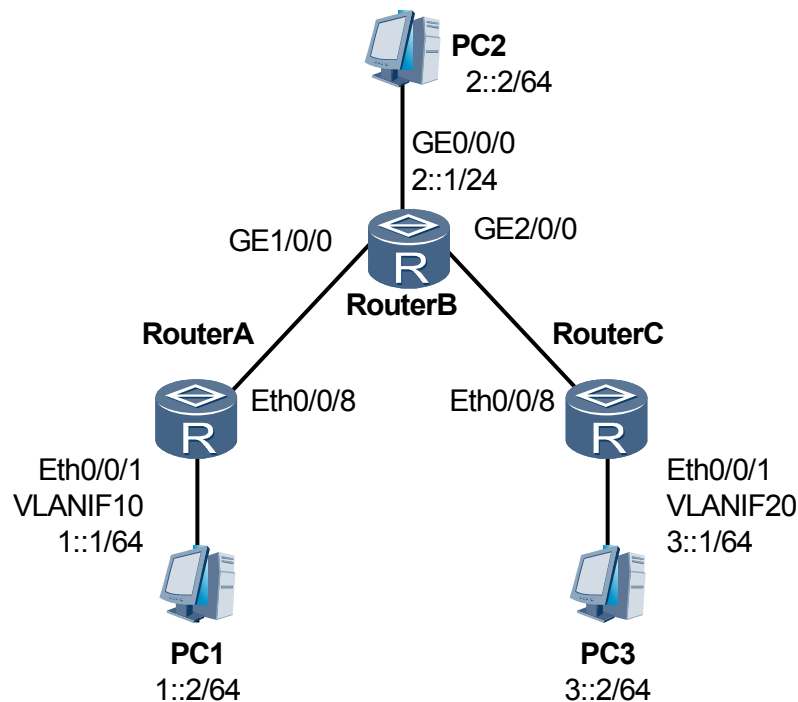
As shown in [Figure 2-2](#), the mask length of all the IPv6 addresses is 64 bits. It is required that every two hosts or routers be interconnected.



NOTE

AR150/200 can be used as RouterA, or RouterC.

Figure 2-2 Networking diagram for configuring IPv6 static routes



In [Figure 2-2](#), PC1, PC2, and PC3 use static routes because they cannot be configured with dynamic routing protocols. The interface of the router connected to the PC uses the IPv6 link-local address.

Configuration Roadmap

The configuration roadmap is as follows:

1. Configure an IPv6 address for each interface on each router for interworking.
2. Configure an IPv6 static route and a default route to the destination address on each router.
3. Configure an IPv6 default gateway on each host to make any two hosts communicate.

Data Preparation

To complete the configuration, you need the following data:

- Link-local addresses of interfaces, as shown in the following table.

Device	Interface	Link-local address
RouterA	Eth0/0/8	FE80::A19:A6FF:FECD:A897
RouterB	GE1/0/0	FE80::E0:FCD5:A2BF:401
RouterB	GE2/0/0	FE80::A19:A6FF:FECD:A896
RouterC	Eth0/0/8	FE80::A19:A6FF:FECD:A895

- Default route on RouterA: its outbound interface Eth0/0/8 and next hop address FE80::E0:FCD5:A2BF:401
- Static route of RouterB: destination address 1:: 64, outbound interface GE1/0/0, and next hop address FE80::A19:A6FF:FECD:A897
- Static route of RouterB: destination address 3:: 64, outbound interface GE2/0/0 and next hop address FE80::A19:A6FF:FECD:A895
- Default route of RouterC: outbound interface Eth0/0/8 and next hop address FE80::A19:A6FF:FECD:A896
- Default gateway addresses of PC1, PC2, and PC3 being 1::1, 2::1, and 3::1 respectively

Procedure

- Step 1** Configure an IPv6 address for each interface.

The configuration details are not described here.

- Step 2** Configure IPv6 static routes.

Configure an IPv6 default route on Router A.

```
[RouterA] ipv6 route-static :: 0 ethernet 0/0/8 FE80::E0:FCD5:A2BF:401
```

Configure two IPv6 static routes on Router B.

```
[RouterB] ipv6 route-static 1:: 64 gigabitethernet 1/0/0 FE80::A19:A6FF:FECD:A897  
[RouterB] ipv6 route-static 3:: 64 gigabitethernet 2/0/0 FE80::A19:A6FF:FECD:A895
```

Configure an IPv6 default route on Router C.

```
[RouterC] ipv6 route-static :: 0 ethernet 0/0/8 FE80::A19:A6FF:FECD:A896
```

- Step 3** Configure host addresses and gateways.

Configure IPv6 addresses for hosts according to the networking diagram, and set default gateway addresses of PC1, PC2, and PC3 to 1::1, 2::1, and 3::1 respectively.

- Step 4** Verify the configuration.

Check the IPv6 routing table of Router A.

```
[RouterA] display ipv6 routing-table  
Routing Table : Public  
Destinations : 5          Routes : 5
```

```

Destination : ::
NextHop     : FE80::E0:FCD5:A2BF:401
Cost       : 0
RelayNextHop : ::
Interface   : Ethernet0/0/8
PrefixLength : 0
Preference  : 60
Protocol    : Static
TunnelID    : 0x0
Flags      : D

Destination : ::1
NextHop     : ::1
Cost       : 0
RelayNextHop : ::
Interface   : InLoopBack0
PrefixLength : 128
Preference  : 0
Protocol    : Direct
TunnelID    : 0x0
Flags      : D

Destination : 1::
NextHop     : 1::1
Cost       : 0
RelayNextHop : ::
Interface   : Vlanif10
PrefixLength : 64
Preference  : 0
Protocol    : Direct
TunnelID    : 0x0
Flags      : D

Destination : 1::1
NextHop     : ::1
Cost       : 0
RelayNextHop : ::
Interface   : Vlanif10
PrefixLength : 128
Preference  : 0
Protocol    : Direct
TunnelID    : 0x0
Flags      : D

Destination : FE80::
NextHop     : ::
Cost       : 0
RelayNextHop : ::
Interface   : NULL0
PrefixLength : 10
Preference  : 0
Protocol    : Direct
TunnelID    : 0x0
Flags      : D
    
```

Run the **ping** command to verify the connectivity.

```

[RouterA] ping ipv6 3::1
PING 3::1 : 56 data bytes, press CTRL_C to break
  Reply from 3::1:
    bytes=56 Sequence=1 hop limit=254 time = 63 ms
  Reply from 3::1:
    bytes=56 Sequence=2 hop limit=254 time = 62 ms
  Reply from 3::1:
    bytes=56 Sequence=3 hop limit=254 time = 62 ms
  Reply from 3::1:
    bytes=56 Sequence=4 hop limit=254 time = 63 ms
  Reply from 3::1:
    bytes=56 Sequence=5 hop limit=254 time = 63 ms
--- 3::1 ping statistics ---
  5 packet(s) transmitted
  5 packet(s) received
  0.00% packet loss
  round-trip min/avg/max = 62/62/63 ms
    
```

Run the **tracert** command to verify the connectivity.

```

[RouterA] tracert ipv6 3::1
traceroute to 3::1 30 hops max, 60 bytes packet
 1 FE80::E0:FCD5:86D4:401 11 ms 3 ms 4 ms
 2 3::1 4 ms 3 ms 3 ms
    
```

----End

Configuration Files

- Configuration file of Router A

```

#
sysname RouterA
#
ipv6
#
    
```

```
vlan batch 10
#
interface Vlanif10
  ipv6 enable
  ipv6 address 1::1/64
#
interface Ethernet0/0/1
  port link-type access
  port default vlan 10
#
interface Ethernet0/0/8
  ipv6 enable
  ipv6 address auto link-local
#
ipv6 route-static :: 0 Ethernet 0/0/8 FE80::E0:FCD5:A2BF:401
#
return
```

● Configuration file of Router B

```
#
sysname RouterB
#
ipv6
#
interface GigabitEthernet0/0/0
  ipv6 address 2::1/64
#
interface GigabitEthernet1/0/0
  ipv6 address auto link-local
#
interface GigabitEthernet2/0/0
  ipv6 address auto link-local
#
ipv6 route-static 1:: 64 GigabitEthernet 1/0/0 FE80::A19:A6FF:FECD:A897
ipv6 route-static 3:: 64 GigabitEthernet 2/0/0 FE80::A19:A6FF:FECD:A895
#
return
```

● Configuration file of Router C

```
#
sysname RouterC
#
ipv6
#
vlan batch 20
#
interface Vlanif20
  ipv6 enable
  ipv6 address 3::1/64
#
interface Ethernet0/0/1
  port link-type access
  port default vlan 20
#
interface Ethernet0/0/8
  ipv6 enable
  ipv6 address auto link-local
#
ipv6 route-static :: 0 Ethernet 0/0/8 FE80::A19:A6FF:FECD:A896
#
return
```

3 RIP Configuration

About This Chapter

RIP can advertise and receive routes to affect the selection of data forwarding paths, and can provide the network management function. RIP is commonly used on small-scale networks.

[3.1 Overview of RIP](#)

RIP is widely used on small-scale network because it is simple to deploy and easier to configure and maintain than OSPF and IS-IS.

[3.2 RIP Features Supported by the AR150/200](#)

The RIP features supported by the AR150/200 include RIPv1, RIPv2, split horizon, poison reverse, and multi-instance.

[3.3 Configuring Basic RIP Functions](#)

To implement RIP features, configure basic RIP functions including enabling RIP, specifying the network segment in which RIP runs, and setting the RIP version.

[3.4 Configuring RIP Route Attributes](#)

By setting RIP route attributes, you can change RIP routing policies to meet the requirements of complex networks.

[3.5 Controlling the Advertising of RIP Routing Information](#)

To meet the requirements of complex networks, accurately controlling the advertising of RIP routing information is essential.

[3.6 Controlling the Receiving of RIP Routing Information](#)

To meet the requirements of complex networks, accurately controlling the receiving of RIP routing information is essential.

[3.7 Configuring RIP-2 Features](#)

Different from RIP-1, RIP-2 supports VLSM, CIDR, and authentication to ensure higher security.

[3.8 Optimizing a RIP Network](#)

You can adjust and optimize the RIP network performance by configuring RIP functions in special network environments, such as configuring RIP timers, setting the interval for sending packets, and setting the maximum number of packets to be sent.

[3.9 Configuring RIP GR](#)

This section describes how to configure RIP GR to avoid incorrect route calculation and packet loss after a RIP router restarts.

[3.10 Configuring BFD for RIP](#)

On a network that runs high-rate data services, BFD for RIP can be configured to quickly detect and respond to network faults.

[3.11 Configuring Static BFD for RIP](#)

BFD provides link failure detection featuring light load and high speed. Static BFD for RIP is a mode to implement the BFD function.

[3.12 Configuring the Network Management Function in RIP](#)

By binding RIP to MIBs, you can view and configure RIP through the NMS.

[3.13 Maintaining RIP](#)

This section describes how to reset RIP connections and clear RIP information.

[3.14 Configuration Examples](#)

In actual networking, RIP versions and whether to import external routes will affect which routes can be learned.

3.1 Overview of RIP

RIP is widely used on small-scale network because it is simple to deploy and easier to configure and maintain than OSPF and IS-IS.

The Routing Information Protocol (RIP) is a simple Interior Gateway Protocol (IGP). RIP is mainly used on small-scale networks such as campus networks and simple regional networks.

RIP uses the distance-vector routing algorithm and exchanges routing information by using User Datagram Protocol (UDP) packets through port 520.

RIP uses the hop count to measure the distance to the destination. The distance is called the routing metric. In RIP, the hop count from a router to its directly connected network is 0, and the hop count from a router to a network, which can be reached through another router, is 1. To speed up route convergence, RIP defines the cost as an integer that ranges from 0 to 15. If the hop count is equal to or exceeds 16, the destination network or host is unreachable because the path is considered to have an infinite metric. It is this limitation to the hop count that makes RIP inapplicable to large-scale networks.

To improve network performance and prevent routing loops, RIP supports both split horizon and poison reverse.

- Split horizon is a method of preventing routing loops in a network and reducing bandwidth consumption. The basic principle is simple: Information about the routing for a particular packet is never sent back in the direction from which it was received.
- Poison reverse is that RIP sets the cost of the route learnt from an interface of a neighbor to 16 (specifying the route as unreachable) and then sends the route from the interface back to the neighbor. In this way, RIP can delete useless routes from the routing table of the neighbor.

RIP has two versions:

- RIPv1
- RIPv2

RIPv1 is a classful routing protocol, whereas RIPv2 is a classless routing protocol. In RIPv2, address 224.0.0.9 is the multicast address of a RIP router.

Compared with RIPv1, RIPv2 has the following advantages:

- Supports route tag and can flexibly control routes on the basis of the tag in the routing policy.
- Provides packets that contain mask information and supports route aggregation and Classless Inter-domain Routing (CIDR).
- Supports the next hop address and can select the optimal next hop address in the broadcast network.
- Uses multicast routes to send update packets. Only RIPv2 routers can receive protocol packets. This reduces the resource consumption.
- To enhance the security, Provides two authentication modes to enhance security: plain-text authentication and MD5 authentication.

3.2 RIP Features Supported by the AR150/200

The RIP features supported by the AR150/200 include RIPv1, RIPv2, split horizon, poison reverse, and multi-instance.

The AR150/200 supports the following RIP features:

- RIPv1 and RIPv2
- RIP multi-instance

3.3 Configuring Basic RIP Functions

To implement RIP features, configure basic RIP functions including enabling RIP, specifying the network segment in which RIP runs, and setting the RIP version.

3.3.1 Establishing the Configuration Task

Before configuring basic RIP functions, familiarize yourself with the applicable environment, complete the pre-configuration tasks, and obtain the required data. This can help you complete the configuration task quickly and accurately.

Applicable Environment

Configuring basic RIP functions allows you to enjoy certain RIP features.

Pre-configuration Tasks

Before configuring basic RIP functions, complete the following tasks:

- Configuring the link layer protocol
- Configuring IP addresses for interfaces to ensure that neighboring nodes are reachable at the network layer

Data Preparation

To configure basic RIP functions, you need the following data.

No.	Data
1	RIP process ID
2	Network segment in which the RIP interface resides
3	RIP version number

3.3.2 Enabling RIP

Creating RIP processes is the prerequisite to performing RIP configurations.

Context

If you run RIP-related commands in the interface view before enabling RIP, the configurations take effect only after RIP is enabled.

Do as follows on the router to be enabled with RIP:

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
rip [ process-id ]
```

The RIP is enabled and the RIP view is displayed.

RIP supports multi-instance. To associate RIP processes with VPN instances, you can run the **rip [process-id] vpn-instance vpn-instance-name** command.

NOTE

For easy management and effective control, RIP supports multi-process and multi-instance. The multi-process feature allows a set of interfaces to be associated with a specific RIP process and an interface can be associated with only one RIP process. This ensures that the specific RIP process performs all the protocol operations only on this set of interfaces. Thus, multiple RIP processes can work on a single router and each process is responsible for a unique set of interfaces. In addition, the routing data is independent between RIP processes; however, routes can be imported between processes.

For the routers that support the VPN, each RIP process is associated with a specific VPN instance. In this case, all the interfaces attached to the RIP process should be associated with the RIP-process-related VPN instance.

----End

3.3.3 Enabling RIP on the Specified Network Segment

After enabling RIP, you need to specify the network segment in which RIP runs. RIP runs only on the interfaces on the specified network segment. RIP does not receive, send, or forward routes on the interfaces that do not reside on the specified network segment.

Context

By default, after RIP is enabled, it is disabled on all interfaces.

Do as follows on the router to be enabled with RIP.

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
rip [ process-id ]
```

The RIP process is enabled and the RIP view is displayed.

Step 3 Run:

```
network network-address
```

RIP is enabled in the specified network segment.

network-address specifies the address of a natural network segment.

 **NOTE**

An interface can be associated with only one RIP process.

If any network segment in which an interface configured with multiple sub-interface IP addresses resides is associated with a RIP process, the interface cannot be associated with any other RIP processes.

----End

3.3.4 Configuring RIP Version Number

RIP versions include RIPv1 and RIPv2. The two versions have different functions.

Context

Do as follows on the RIP router.

Procedure

- Configuring the Global RIP Version Number
 1. Run:

```
system-view
```

The system view is displayed.
 2. Run:

```
rip [ process-id ]
```

The RIP process is enabled and the RIP view is displayed.
 3. Run:

```
version { 1 | 2 }
```

The global RIP version number is specified.
- Configuring the RIP Version Number for an Interface
 1. Run:

```
system-view
```

The system view is displayed.
 2. Run:

```
interface interface-type interface-number
```

The interface view is displayed.
 3. Run:

```
rip version { 1 | 2 [ broadcast | multicast ] }
```

The RIP version number of the packets received by the interface is specified.

 **NOTE**

By default, an interface receives both RIPv1 and RIPv2 packets but sends only RIPv1 packets. When configuring RIPv2 on an interface, you can specify the mode in which the interface sends packets. If no RIP version number is configured in the interface view, the global RIP version is used. The RIP version set on an interface takes precedence over the global RIP version.

---End

3.3.5 Checking the Configuration

After basic RIP functions are successfully configured, you can view the current running status, configuration, and routing information of RIP.

Prerequisites

The configurations of basic RIP functions are complete.

Procedure

- Run **display rip** [*process-id* | **vpn-instance** *vpn-instance-name*] command to check the running status and configuration of RIP.
- Run **display rip process-id route** command to check all the RIP routes that are learned from other routers.
- Run **display default-parameter rip** command to check the default RIP configuration.
- Run the **display rip process-id statistics interface** { **all** | *interface-type interface-number* [**verbose** | **neighbor** *neighbor-ip-address*] } command to check statistics about RIP interfaces.

---End

3.4 Configuring RIP Route Attributes

By setting RIP route attributes, you can change RIP routing policies to meet the requirements of complex networks.

3.4.1 Establishing the Configuration Task

RIP route attributes include the RIP preference, additional metrics of an interface, and maximum number of equal-cost routes.

Applicable Environment

For complex networks, you can set RIP route attributes to change RIP routing policies. After performing the configuration procedures in this section, you can:

- Affect route selection by changing the additional metric of a RIP interface.
- Change the matching order by configuring the RIP preference when multiple routing protocols discover routes to the same destination.
- Implement load balancing among multiple equal-cost routes.

Pre-configuration Tasks

Before configuring RIP route attributes, complete the following tasks:

- Configuring IP addresses for interfaces to ensure that neighboring nodes are reachable at the network layer
- **Configuring Basic RIP Functions**

Data Preparation

To configure RIP route attributes, you need the following data.

No.	Data
1	Additional metric of the interface
2	RIP preference
3	Maximum number of equal-cost routes

3.4.2 Configuring Additional Metrics of an Interface

The additional metric is the metric (hop count) to be added to the original metric of a RIP route. You can specify commands to set additional metrics for incoming and outgoing RIP routes.

Context

The additional metric is added to the original metric of the RIP route.

- The **rip metricin** command is used to add an additional metric to an incoming route. After this route is added to the routing table, its metric in the routing table changes. Running this command affects route selection on the local device and other devices on the network.
- The **rip metricout** command is used to add an additional metric to an outgoing route. When this route is advertised, an additional metric is added to this route, but the metric of the route in the routing table does not change. Running this command does not affect route selection on the local device or other devices on the network.

Do as follows on the RIP router:

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
interface interface-type interface-number
```

The interface view is displayed.

Step 3 Run:

```
rip metricin value
```

The metric added to an incoming route is set.

Step 4 Run:

```
rip metricout { value | { acl-number | acl-name acl-name | ip-prefix ip-prefix-name } value1 }
```

The metric added to an outgoing route is set.

 **NOTE**

You can specify the value of the metric to be added to the RIP route that passes the filtering policy by specifying *value1* through an ACL or an IP prefix list. If a RIP route does not pass the filtering, its metric is increased by 1.

----End

3.4.3 Configuring RIP Preference

When there are routes discovered by multiple routing protocols on the same Router, you can set RIP preferences to instruct the Router to prefer certain RIP routes over others.

Context

Do as follows on the RIP router:

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
rip [ process-id ]
```

The RIP process is enabled and the RIP view is displayed.

Step 3 Run:

```
preference { preference | route-policy route-policy-name } *
```

The RIP preference is set.

By default, the RIP preference is 100.

----End

3.4.4 Setting the Maximum Number of Equal-Cost Routes

By setting the maximum number of equal-cost RIP routes, you can change the number of routes for load balancing.

Context

Do as follows on the RIP router:

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
rip [ process-id ]
```

The RIP process is enabled and the RIP view is displayed.

Step 3 Run:

```
maximum load-balancing number
```

The maximum number of equal-cost routes is set.

---End

3.4.5 Checking the Configuration

After RIP route attributes are successfully set, you can view the current running status, configuration, and routing information about RIP.

Prerequisites

The configurations for RIP route attributes are complete.

Procedure

- Run **display rip** [*process-id* | **vpn-instance** *vpn-instance-name*] command to check the running status and configuration of RIP.
- Run **display rip process-id database** command to check all activated routes in the RIP database.
- Run **display rip process-id route** command to check all the RIP routes that are learned from other routers.

---End

3.5 Controlling the Advertising of RIP Routing Information

To meet the requirements of complex networks, accurately controlling the advertising of RIP routing information is essential.

3.5.1 Establishing the Configuration Task

RIP routing information can be advertised through default routes, Update packets, and imported external routes.

Applicable Environment

To meet the requirements of a network, you need to control the advertising of RIP routing information accurately. After performing the configuration procedures in this section, you can:

- Advertise default routes to neighbors.
- Suppress interfaces from sending RIP Update packets.
- Import external routes from various routing protocols and filter the routes to be advertised.

Pre-configuration Tasks

Before configuring the router to control the advertising of RIP routing information, complete the following tasks:

- Configuring IP addresses for interfaces to ensure that neighboring nodes are reachable at the network layer
- **Configuring Basic RIP Functions**

Data Preparation

To control the advertising of RIP routing information, you need the following data.

No.	Data
1	Metric of the default route to be advertised
2	Number of the interface that is suppressed from sending RIP Update packets
3	Protocol name and process ID of the external route to be imported

3.5.2 Configuring RIP to Advertise Default Routes

A default route is a route destined for 0.0.0.0. By default, RIP does not advertise default routes to its neighbors.

Context

Do as follows on the RIP router:

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
rip [ process-id ]
```

The RIP process is enabled and the RIP view is displayed.

Step 3 Run:

```
default-route originate [ match default [ avoid-learning ] ] [ cost cost ]
```

RIP is configured to advertise a default route.

You can configure a router to advertise a default route or set the default routes in routing table with the specified metric to its RIP neighbors.

----End

3.5.3 Disabling an Interface from Sending Update Packets

Disabling interfaces from sending Update packets is a method of preventing routing loops and can be implemented in two ways.

Context

Do as follows on the RIP router:

Procedure

- Configuration in a RIP Process (with a High Priority)

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
rip [ process-id ]
```

The RIP process is enabled and the RIP view is displayed.

3. Run one of the following commands depending on the site requirements.

Run:

```
silent-interface all
```

All interfaces are disabled from sending Update packets.

Run:

```
silent-interface interface-type interface-number
```

An interface is disabled from sending Update packets.

You can set an interface to silent so that it only receives Update packets to update its routing table. The **silent-interface** command takes precedence over the **rip output** command in the interface view.

By default, an interface can receive and send Update packets.

- Configuration in the Interface View (with a Low Priority)

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
interface interface-type interface-number
```

The interface view is displayed.

3. Run:

```
undo rip output
```

The interface is disabled from sending RIP Update packets.

By running this command, you can specify whether to send RIP Update packets on an interface. The **silent-interface** command takes precedence over the **undo rip output** command. By default, an interface is allowed to send RIP Update packets.

----End

3.5.4 Configuring RIP to Import External Routes

To enrich its routing information, RIP can import the routes learned by other processes or other routing protocols.

Context

Do as follows on the RIP router:

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
rip [ process-id ]
```

The RIP process is enabled and the RIP view is displayed.

Step 3 (Optional) Run:

```
default-cost cost
```

The default cost of imported routes is set.

If no cost is specified when external routes are imported, the default cost is used.

Step 4 Run:

```
import-route bgp [ cost { cost | transparent } | route-policy route-policy-name ]  
* or import-route { { static | direct } | { { rip | ospf | isis } [ process-id ] } }  
[ cost cost | route-policy route-policy-name ] *
```

Step 5 (Optional) Run:

```
filter-policy { acl-number | acl-name acl-name | ip-prefix ip-prefix-name } export  
[ protocol [ process-id ] | interface-type interface-number ]
```

The imported routes are filtered when being advertised.

If the routing information to be advertised by RIP contains the routes imported from other routing protocols, you can specify *protocol* to filter the specified routes. If *protocol* is not specified, all the routing information to be advertised will be filtered, including the imported routes and local RIP routes (directly connected routes).

NOTE

The Tag field in RIP is 16 bits in length, whereas the Tag field in other routing protocols is 32 bits in length. If the routes of other routing protocols are imported and the tag is used in the routing policy, ensure that the tag value does not exceed 65535. Otherwise, the routing policy becomes invalid or the matching result is incorrect.

----End

3.5.5 Checking the Configuration

After the function of controlling the advertising of RIP routing information is successfully configured, you can view the current running status, configuration, and routing information about RIP.

Prerequisites

The configurations for controlling the advertising of RIP routing information are complete.

Procedure

- Run the **display rip** [*process-id* | **vpn-instance** *vpn-instance-name*] command to check the running status and configuration of RIP.
- Run the **display rip process-id database** command to check all activated routes in the RIP database.
- Run the **display rip process-id route** command to check all the RIP routes that are learned from other routers..

----End

3.6 Controlling the Receiving of RIP Routing Information

To meet the requirements of complex networks, accurately controlling the receiving of RIP routing information is essential.

3.6.1 Establishing the Configuration Task

You can obtain RIP routing information by receiving Update packets and host routes.

Applicable Environment

In practice, to meet the requirements of a complex network, it is required to control the receiving of RIP routing information accurately. After performing configuration procedures in this section, you can:

- Disable an interface from receiving RIP Update packets.
- Filter the received routing information.
- Import external routes from various routing protocols and filter the imported routes.

Pre-configuration Tasks

Before configuring a router to control the receiving of RIP routing information, complete the following tasks:

- Configuring IP addresses for interfaces to ensure that neighboring nodes are reachable at the network layer
- [Configuring Basic RIP Functions](#)

Data Preparation

To control the receiving of RIP routing information, you need the following data.

No.	Data
1	ACL used to filter the routing information

3.6.2 Disabling an Interface from Receiving RIP Update Packets

Disabling interfaces from receiving Update packets is a method of preventing routing loops.

Context

By default, an interface is allowed to receive RIP Update packets.

Do as follows on the RIP router:

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
interface interface-type interface-number
```

The interface view is displayed.

Step 3 Run:

```
undo rip input
```

The interface is disabled from receiving RIP Update packets.

----End

3.6.3 Disabling RIP from Receiving Host Routes

When you disable RIP from receiving host routes on a router, the router rejects to receive host routes. This prevents the router from receiving a large number of unnecessary routes and thus avoiding wasting network resources.

Context

In certain situations, a router may receive a large number of host routes from the same network segment. These routes are not required in route addressing, but consume many network resources. You can configure the router to refuse to accept host routes by disabling RIP from accepting host routes.

By default, host routes are added to the routing table.

Do as follows on the RIP router:

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
rip [ process-id ]
```

The RIP process is enabled and the RIP view is displayed.

Step 3 Run:

```
undo host-route
```

RIP is disabled from adding host routes to the routing table.

 **NOTE**

undo host-route command will not be effective in RIP version 2. By default, RIP version 2 always supports host-route.

----End

3.6.4 Configuring RIP to Filter the Received Routes

By specifying ACLs and IP prefix lists, you can configure the inbound policy to filter the routes to be received. You can also configure a router to receive only RIP packets from a specified neighbor.

Context

The router can filter routing information. To filter the imported and advertised routes, you can configure inbound and outbound routing policies by specifying ACLs and IP prefix lists.

You can also configure the router to receive RIP packets only from a specified neighbor.

Do as follows on the RIP router:

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
rip [ process-id ]
```

The RIP process is enabled and the RIP view is displayed.

Step 3 Depending on type of desired filtering, run one of following commands to configure RIP to filter the received routes:

● Run:

```
filter-policy { acl-number | acl-name acl-name } import
```

The learned routing information is filtered based on an ACL.

● Run:

```
filter-policy gateway ip-prefix-name import
```

The routing information advertised by neighbors is filtered based on the IP prefix list.

● Run:

```
filter-policy ip-prefix ip-prefix-name [ gateway ip-prefix-name ] import  
[ interface-type interface-number ]
```

The routes learned by the specified interface are filtered based on the IP prefix list and neighbors.

 **NOTE**

To filter routes to be advertised, run the **filter-policy { acl-number | acl-name acl-name | ip-prefix ip-prefix-name } export [protocol [process-id] | interface-type interface-number]** command.

----End

3.6.5 Checking the Configuration

After the receiving of RIP routing information is successfully controlled, you can view the current running status, configuration, and routing information about RIP.

Procedure

- Run the **display rip** [*process-id* | **vpn-instance** *vpn-instance-name*] command to check the running status and configuration of RIP.
- Run the **display rip process-id database** [**verbose**] command to check all activated RIP routes in the database.
- Run the **display rip process-id interface** [*interface-type interface-number*] [**verbose**] command to check information about the RIP interface.
- Run the **display rip process-id neighbor** [**verbose**] command to check information about RIP neighbors.
- Run the **display rip process-id route** command to check all the RIP routes that are learned from other routers.

----End

3.7 Configuring RIP-2 Features

Different from RIP-1, RIP-2 supports VLSM, CIDR, and authentication to ensure higher security.

3.7.1 Establishing the Configuration Task

Before configuring RIP-2 features, familiarize yourself with the applicable environment, complete the pre-configuration tasks, and obtain the data required for the configuration. This will help you complete the configuration task quickly and accurately.

Applicable Environment

RIP-2 is a type of classless routing protocol. A RIP-2 packet carries subnet mask information. Deploying a RIP-2 network saves IP addresses. For a network on which the IP addresses of devices are not consecutive, only RIP-2 can be deployed, whereas RIP-1 cannot be deployed.

RIP-2 features include:

- RIP-2 route summarization
- RIP-2 authentication mode

Pre-configuration Tasks

Before configuring RIP-2 features, complete the following tasks:

- Configuring the link layer protocol
- Configuring IP addresses for interfaces to ensure that neighboring nodes are reachable at the network layer

Data Preparation

To configure RIP-2 features, you need the following data.

No.	Data
1	RIP-2 process ID
2	Network segment where the RIP-2 interface resides

3.7.2 Configuring RIP-2 Route Summarization

Route summarization is enabled in RIP-1 by default, and no need to be configured. RIP-2 supports VLSM and CIDR. You can configure route summarization in RIP-2 to improve the flexibility of RIP-2. To broadcast all subnet routes, you can disable route summarization in RIP-2.

Context

Route summarization indicates that multiple subnet routes on the same natural network segment are summarized into one route with the natural mask when being advertised to other network segments. Therefore, route summarization reduces the network traffic and the size of the routing table.

Route summarization does not take effect in RIP-1. RIP-2 supports Variable Length Subnet Mask (VLSM) and Classless Interdomain Routing (CIDR). To broadcast all subnet routes, you can disable automatic route summarization of RIP-2.

Do as follows on the RIP router:

NOTE

Route summarization is invalid when split horizon or poison reverse is configured. When the summarized routes are sent outside the natural network boundary, split horizon or poison reverse in related views needs to be disabled.

Procedure

- Enabling RIP-2 Automatic Route Summarization
 1. Run:

```
system-view
```

The system view is displayed.
 2. Run:

```
rip [ process-id ]
```

The RIP process is enabled and the RIP view is displayed.
 3. Run:

```
version 2
```

RIP-2 is configured.
 4. Run:

```
summary [ always ]
```

- Enable the RIP-2 automatic route summarization when split horizon is disabled, there is no need to configure **always**.
- Enable the RIP-2 automatic route summarization irrespective of split horizon configuration, **always** must be configured.

 **NOTE**

The **summary** command is used in the RIP view to enable classful network-based route summarization.

- Configuring RIP-2 to Advertise the Summary Address

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
interface interface-type interface-number
```

The interface view is displayed.

3. Run:

```
rip summary-address ip-address mask [ avoid-feedback ]
```

The local summary address of RIP-2 is advertised.

 **NOTE**

The **rip summary-address** *ip-address mask* [**avoid-feedback**] command is run in the interface view to enable classless network-based route summarization.

---End

3.7.3 Configuring Packet Authentication of RIP-2

RIP-2 supports the ability to authenticate protocol packets and provides two authentication modes, Simple authentication and Message Digest 5 (MD5) authentication, to enhance security.

Context

RIP-2 supports two authentication modes:

- Simple authentication
- MD5 authentication

In simple authentication mode, the unencrypted authentication key is sent in every RIP-2 packet. Therefore, simple authentication does not guarantee security, and cannot meet the requirements for high security.

Do as follows on the RIP router:

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
interface interface-type interface-number
```

The interface view is displayed.

Step 3 Run the following command as required:

- Run:

```
rip authentication-mode simple { [ plain ] plain-text | cipher password-key }
```

Simple authentication is configured for RIP-2 packets.

- Run:

```
rip authentication-mode usual { plain plain-text | [ cipher ] password-key }
```

MD5 usual authentication is configured for RIP-2 packets.

- Run:

```
rip authentication-mode nonstandard { keychain keychain-name | { { plain plain-text | [ cipher ] password-key } key-id } }
```

MD5 nonstandard authentication is configured for RIP-2 packets.

NOTE

The MD5 type must be specified if MD5 authentication is configured. The **usual** type supports private standard authentication packets, and the **nonstandard** type supports IETF standard authentication packets. The MD5 authentication password that starts and ends with \$@\$@ is invalid, because \$@\$@ is used to distinguish old and new passwords.

----End

3.7.4 Checking the Configuration

After RIP-2 features are successfully configured, you can view the current running status, configuration, and routing information of RIP.

Prerequisites

The configurations of RIP-2 features are complete.

Procedure

- Run the **display rip** [*process-id* | **vpn-instance** *vpn-instance-name*] command to check the running status and configuration of RIP.
- Run the **display rip process-id database** [**verbose**] command to check all activated RIP routes in the database.
- Run the **display rip process-id route** command to check all the RIP routes that are learned from other routers.

----End

3.8 Optimizing a RIP Network

You can adjust and optimize the RIP network performance by configuring RIP functions in special network environments, such as configuring RIP timers, setting the interval for sending packets, and setting the maximum number of packets to be sent.

3.8.1 Establishing the Configuration Task

Before adjusting and optimizing the RIP network performance, familiarize yourself with the applicable environment, complete the pre-configuration tasks, and obtain the data required for the configuration. This will help you complete the configuration task quickly and accurately.

Applicable Environment

On certain networks, you need to configure RIP features and optimize the performance of a RIP network. After performing configuration procedures in this section, you can:

- Change the convergence speed of the RIP network by adjusting the values of RIP timers.
- Reduce the consumption of device resources and network bandwidth by adjusting the number of packets to be sent by interfaces and the interval at which packets are sent.
- Configure split horizon or poison reverse to prevent routing loops.
- After the replay-protect function is enabled, neighbors can communicate after a RIP process is restarted.
- Check the validity of packets and authenticate packets on a network demanding high security.
- Run RIP on a link that does not support broadcast or multicast packets.

Pre-configuration Tasks

Before optimizing a RIP network, complete the following tasks:

- Configuring IP addresses for interfaces to ensure that neighboring nodes are reachable at the network layer
- [Configuring Basic RIP Functions](#)

Data Preparation

To optimize a RIP network, you need the following data.

No.	Data
1	Values of timers
2	Number of Update packets that an interface sends each time and interval for sending an Update packet
3	Maximum number of equal-cost routes
4	Packet authentication mode and password
5	IP addresses of RIP neighbors

3.8.2 Configuring RIP Timers

RIP has three timers: Update timer, Age timer and Garbage-collect timer. Changing the values of the three timers affects the RIP convergence speed.

Context

RIP has three timers: Update timer, Age timer and Garbage-collect timer. Changing the values of the three timers affects the RIP convergence speed. For details on timers, see corresponding description in the chapter "RIP" in the *Huawei AR150&200 Series Enterprise Routers Feature Description - IP Routing*.

Do as follows on the RIP router:

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
rip [ process-id ]
```

The RIP process is enabled and the RIP view is displayed.

Step 3 Run:

```
timers rip update age garbage-collect
```

RIP timers are configured.

NOTE

- RIP timers take effect immediately after being changed.
- Route flapping occurs if the values of the times are set improperly. The relationship between the values is as follows: *update* must be smaller than *age* and *update* must be smaller than *garbage-collect*. For example, if the update time is longer than the aging time, and a RIP route changes within the update time, the router cannot inform its neighbors of the change on time.
- You must configure RIP timers based on the network performance and uniformly on all the routers running RIP. This avoids unnecessary network traffic or route flapping.

By default, the Update timer is 30s; the Age timer is 180s; the Garbage-collect timer is four times the Update timer, namely, 120s.

In practice, the Garbage-collect timer is not fixed. If the Update timer is set to 30s, the Garbage-collect timer may range from 90s to 120s.

Before permanently deleting an unreachable route from the routing table, RIP advertises this route (with the metric being set to 16) by periodically sending Update packets four times. Subsequently, all the neighbors know that this route is unreachable. Because a route may not always become unreachable at the beginning of an Update period, the Garbage-collect timer is actually three or four times the Update timer.

---End

3.8.3 Setting the Interval for Sending Packets and the Maximum Number of the Sent Packets

By setting the interval for sending RIP Update packets and the maximum number of Update packets to be sent each time, you can effectively control the memory used by a Router to process RIP Update packets.

Context

Do as follows on the RIP router:

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
interface interface-type interface-number
```

The interface view is displayed.

Step 3 Run:

```
rip pkt-transmit { interval interval | number pkt-count } *
```

The interval for sending Update packets and the maximum number of packets sent each time are set on the interface.

---End

3.8.4 Configuring Split Horizon and Poison Reverse

You can configure split horizon and poison reverse to prevent routing loops.

Context

If both split horizon and poison reverse are configured, only poison reverse takes effect.

On Non-Broadcast Multi-Access (NBMA) networks such as frame relay (FR) and X.25 networks, split horizon is disabled by default.

Do as follows on the RIP router:

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
interface interface-type interface-number
```

The interface view is displayed.

Step 3 Run the following command as required:

● Run:

```
rip split-horizon
```

Split horizon is enabled.

● Run:

```
rip poison-reverse
```

Poison reverse is enabled.

---End

3.8.5 Enabling replay-protect Function

By enabling the replay-protect function, you can obtain the Identification field in the last RIP packet sent by a RIP interface before it goes Down. This prevents RIP routing information on both ends from being unsynchronized or lost.

Context

If the Identification field in the last RIP packet sent before a RIP interface goes Down is X, after the interface goes Up, the Identification field in the subsequent RIP packet sent by this interface becomes 0. If the remote end does not receive the RIP packet with the Identification field being 0, subsequent RIP packets will be discarded until the remote end receives the RIP packet with the Identification field being X+1. This leads to the unsynchronization and loss of RIP routing information of both ends.

To solve this problem, you need to enable the replay-protect function so that RIP can obtain the Identification field in the last RIP packet sent before the RIP interface goes Down and increase the Identification field in the subsequent RIP packet by one.

Do as follows on the RIP router:

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
interface interface-type interface-number
```

The interface view is displayed.

Step 3 Run:

```
rip authentication-mode md5 nonstandard password-key key-id
```

RIPv2 is configured to use MD5 authentication, and authentication packets use the nonstandard packet format.

 **NOTE**

Before running the **rip replay-protect** command, run the **rip authentication-mode md5 nonstandard** command in the RIP interface view to configure MD5 authentication packets to use the nonstandard packet format (private standard).

Step 4 Run:

```
rip replay-protect
```

The replay-protect function is enabled.

 **NOTE**

- For details of the Identification field in an IP packet, see *Feature Description - IP Services*.
- If you run the **rip replay-protect** command in the same view multiple times, only the last configuration takes effect.

---End

3.8.6 Configuring RIP to Check the Validity of Update Packets

The check on RIP Update packets includes the check on zero fields in RIPv1 packets and the check on source addresses of RIP Update packets. The two types of check have different functions and applications.

Context

Do as follows on the RIP router:

Procedure

- Configuring the Zero Field Check for RIPv1 Packets

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
rip [ process-id ]
```

The RIP process is enabled and the RIP view is displayed.

3. Run:

```
checkzero
```

The zero field check is configured for RIPv1 packets.

Certain fields in a RIPv1 packet must be 0s, and these fields are called zero fields. RIPv1 checks the zero fields on receiving a packet. If the value of any zero field in a RIPv1 packet is not 0, this packet is not processed.

As a RIPv2 packet does not contain any zero field, configuring the zero field check is invalid in RIPv2.

- Configuring the Source Address Check for RIP Update Packets

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
rip [ process-id ]
```

The RIP process is enabled and the RIP view is displayed.

3. Run:

```
verify-source
```

The source address check is configured for RIP Update packets.

When receiving a packet, RIP checks the source address of the packet. If the packet fails in the check, it is not processed.

By default, the source address check is enabled.

----End

3.8.7 Configuring RIP Neighbors

Generally, RIP sends packets by using broadcast or multicast addresses. To run RIP on the links that do not support the forwarding of broadcast or multicast packets, you need to specify RIP neighbors.

Context

Generally, RIP sends packets by using broadcast or multicast addresses. If RIP needs to run on the links that do not support the forwarding of broadcast or multicast packets, you need to configure the devices at both ends of the link as each other's neighbor.

Do as follows on the RIP router:

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
rip [ process-id ]
```

The RIP process is enabled and the RIP view is displayed.

Step 3 Run:

```
peer ip-address
```

The RIP neighbor is configured.

----End

3.8.8 Checking the Configuration

After the function of adjusting and optimizing the RIP network performance is successfully configured, you can view the current running status, routing information, neighbor information, and interface information of RIP.

Prerequisites

The configurations of optimizing a RIP network are complete.

Procedure

- Run the **display rip** [*process-id* | **vpn-instance** *vpn-instance-name*] command to check the running status and configuration of RIP.
- Run the **display rip process-id database** [**verbose**] command to check all activated RIP routes in the database.
- Run the **display rip process-id interface** [*interface-type interface-number*] [**verbose**] command to check information about the RIP interface.

- Run the **display rip process-id neighbor [verbose]** command to check information about RIP neighbors.
- Run the **display rip process-id route** command to check all the RIP routes that are learned from other routers.

---End

3.9 Configuring RIP GR

This section describes how to configure RIP GR to avoid incorrect route calculation and packet loss after a RIP router restarts.

3.9.1 Establishing the Configuration Task

In practice, you can configure RIP GR on the device with two main control boards to prevent service forwarding from being affected by the fault on one main control board.

Applicable Environment

To avoid traffic interruption and route flapping caused by master/slave switchover, you can enable RIP graceful restart (GR). GR is a technology used to ensure normal traffic forwarding and non-stop forwarding of key services during the restart of routing protocols.

After a RIP process is restarted through GR, the Restarter and the Helper re-establish the neighbor relationship and update the routing table and forwarding table. This ensures non-stop traffic forwarding and stabilizes the network topology. During RIP GR, except the neighbor of the device where master/slave switchover occurs, other routers do not detect the route change.

NOTE

In practice, you can configure RIP GR on the device with two main control boards to prevent service forwarding from being affected by the fault on one main control board.

The AR150/200 can function as only the Helper router, but cannot function as the Restarter router.

Pre-configuration Tasks

Before configuring RIP GR, complete the following tasks:

- Configuring IP addresses for interfaces to ensure that neighboring nodes are reachable at the network layer
- **Configuring basic RIP functions**, establish the neighbor relationship successfully

Data Preparation

To configure RIP GR, you need the following data

No.	Data
1	RIP process ID
2	Parameters for establishing a GR session

3.9.2 Enabling RIP GR

To avoid traffic interruption and route flapping caused by master/slave switchover, you can enable RIP GR.

Context

Do as follows on the router to be enabled with GR:

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
rip [ process-id ]
```

The RIP view is displayed.

Step 3 Run:

```
graceful-restart [ period period | wait-time time | planned-only time ] *
```

RIP GR is enabled.

When most routers on a network do not support RIP GR, setting **wait-time time** to a greater value is recommended. This ensures that the Restarter has enough time to learn correct routes.

----End

Follow-up Procedure

If the Restarter finishes GR within the GR period specified by **period period**, the Restarter automatically exits from GR. Otherwise, the Restarter is forced to exit from GR.

3.9.3 Checking the Configuration

After RIP GR is configured, you can check the RIP GR status.

Prerequisites

The configurations of RIP GR are complete.

Procedure

- Run the **display rip process-id graceful-restart [verbose]** command to check the status of RIP GR.

----End

3.10 Configuring BFD for RIP

On a network that runs high-rate data services, BFD for RIP can be configured to quickly detect and respond to network faults.

Applicable Environment

Generally, RIP uses timers to receive and send Update messages to maintain neighbor relationships. If a RIP device does not receive an Update message from a neighbor after the Age timer expires, the RIP device will announce that this neighbor goes Down. The default value of the Age timer is 180s. If a link fault occurs, RIP can detect this fault after 180s. If high-rate data services are deployed on a network, a great deal of data will be lost during the aging time.

BFD provides millisecond-level fault detection. It can rapidly detect faults in protected links or nodes and report them to RIP. This speeds up RIP processes's response to network topology changes and achieves rapid RIP route convergence.

In BFD for RIP, BFD session establishment is triggered by RIP. When establishing a neighbor relationship, RIP will send detection parameters of the neighbor to BFD. Then, a BFD session will be established based on these detection parameters. If a link fault occurs, the local RIP process will receive a neighbor unreachable message within seconds. Then, the local RIP device will delete routing entries in which the neighbor relationship is Down and use the backup path to transmit messages.

Either of the following methods can be used to configure BFD for RIP:

- **Enable BFD in a RIP process:** This method is recommended when BFD for RIP needs to be enabled on most RIP interfaces.
- **Enable BFD on RIP interfaces:** This method is recommended when BFD for RIP needs to be enabled on a small number of RIP interfaces.

NOTE

A BFD session currently does not detect route switching. If the change of bound peer IP address causes a route to switch to another link, the BFD session is negotiated again only when the original link fails.

Pre-configuration Tasks

Before configuring BFD for RIP, complete the following tasks:

- Assigning an IP address to each interface to ensure reachability between neighboring nodes at the network layer
- **Configuring Basic RIP Functions**

Data Preparation

To complete the configuration, you need the following data.

No.	Data
1	ID of a RIP process to be enabled with BFD
2	Type and number of an interface to be enabled with BFD
3	(Optional) BFD session parameter values NOTE Default BFD session parameter values are recommended.

Procedure

- Enable BFD in a RIP process.
 1. Run:
`system-view`
The system view is displayed.
 2. Run:
`bfd`
BFD is enabled globally.
 3. Run:
`quit`
Return to the system view.
 4. Run:
`rip process-id`
The RIP view is displayed.
 5. Run:
`bfd all-interfaces enable`
BFD is enabled in the RIP process to establish a BFD session.

If BFD is enabled globally, RIP will use default BFD parameters to establish BFD sessions on all the interfaces where RIP neighbor relationships are in the Up state.
 6. (Optional) Run:
`bfd all-interfaces { min-rx-interval min-receive-value | min-tx-interval min-transmit-value | detect-multiplier detect-multiplier-value } *`
The values of BFD parameters used to establish the BFD session are set.

BFD parameter values are determined by the actual network situation and network reliability requirement.
 - If links have a high reliability requirement, reduce the interval at which BFD packets are sent.
 - If links have a low reliability requirement, increase the interval at which BFD packets are sent.Running the `bfd all-interfaces` command changes BFD session parameters on all RIP interfaces. The default detection multiplier and interval at which BFD packets are sent are recommended.
 7. (Optional) Perform the following operations to prevent an interface in the RIP process from establishing a BFD session:
 - Run the `quit` command to return to the system view.
 - Run the `interface interface-type interface-number` command to enter the view of a specified interface.
 - Run the `rip bfd block` command to prevent the interface from establishing a BFD session.
- Enable BFD on RIP interfaces.
 1. Run:
`system-view`

- The system view is displayed.
- Run:
`bfd`
BFD is enabled globally.
 - Run:
`quit`
Return to the system view.
 - Run:
`interface interface-type interface-number`
The view of the specified interface is displayed.
 - Run:
`rip bfd enable`
BFD is enabled on the interface to establish a BFD session.
 - (Optional) Run:
`rip bfd { min-rx-interval min-receive-value | min-tx-interval min-transmit-value | detect-multiplier detect-multiplier-value } *`
The values of BFD parameters used to establish the BFD session are set.

----End

Checking the Configuration

After enabling BFD for RIP at both ends of a link, run the `display rip bfd session { interface interface-type interface-number | neighbor-id | all }` command. You can see that the BFDState field value on the local router is displayed Up.

3.11 Configuring Static BFD for RIP

BFD provides link failure detection featuring light load and high speed. Static BFD for RIP is a mode to implement the BFD function.

Context

Establishing BFD sessions between RIP neighbors can rapidly detect faults on links and speed up response of RIP to network topology changes. Static BFD implements the following functions:

- One-arm BFD: If some devices on a network support BFD but some do not, configure one-arm BFD to implement fault detection.
- Two-arm BFD: If all the devices on a network support BFD, configure two-arm BFD to implement fault detection.

Static BFD must be enabled using a command and session parameters are also set using commands.

Pre-configuration Tasks

Before configuring static BFD for RIP, complete the following tasks:

- Assigning an IP address to each interface to ensure IP connectivity
- **Configuring basic RIP functions**

Data Preparation

To complete the configuration, you need the following data:

No.	Data
1	ID of a RIP process
2	Type and number of the interface to be enabled with BFD

Procedure

Step 1 Enable BFD globally.

1. Run:

```
system-view
```

The system view is displayed.
2. Run:

```
bfd
```

BFD is enabled globally.
3. Run:

```
quit
```

Return to the system view.

NOTE

To configure one-arm BFD, go to Step 2. To configure two-arm BFD, go to Step 3.

Step 2 Configure one-arm BFD.

1. Run:

```
bfd cfg-name bind peer-ip peer-ip interface interface-type interface-number  
one-arm-echo
```

BFD is enabled between the specified interface and peer router.

If a peer IP address and a local interface are specified, BFD detects only a single-hop link, that is, a route with the interface specified in the **bfd** command as the outbound interface and with the peer IP address specified in the **peer-ip** command as the next-hop address.
2. Run:

```
discriminator local discr-value
```

The local discriminator is set.
3. (Optional) Run:

```
min-echo-rx-interval interval
```

The minimum interval at which BFD packets are received is configured.
4. Run:

```
commit
```

The configuration is committed.

5. Run:

```
quit
```

Return to the system view.

Step 3 Configure two-arm BFD.

1. Run:

```
bfd cfg-name bind peer-ip ip-address [ interface interface-type interface-  
number ]
```

BFD binding is created.

If a peer IP address and a local interface are specified, BFD detects only a single-hop link, that is, a route with the interface specified in the **bfd** command as the outbound interface and with the peer IP address specified in the **peer-ip** command as the next-hop address.

2. Set discriminators.

- Run:

```
discriminator local discr-value
```

The local discriminator is set.

- Run:

```
discriminator remote discr-value
```

The remote discriminator is set.

The local discriminator must be the remote discriminator of the device on the other end; otherwise, a BFD session cannot be established. The local and remote discriminators cannot be modified after being configured.

NOTE

local *discr-value* set on the local device is the same as that of **remote** *discr-value* set on the remote device. **remote** *discr-value* set on the local device is the same as that of **local** *discr-value* set on the remote device.

3. Run:

```
commit
```

The configuration is committed.

4. Run:

```
quit
```

Return to the system view.

Step 4 Enable static BFD on an interface.

1. Run:

```
interface interface-type interface-number
```

The view of the specified interface is displayed.

2. Run:

```
rip bfd static
```

Static BFD is enabled on the interface.

3. Run:

```
quit
```

Return to the system view.

---End

Checking the Configuration

After configuring static BFD for RIP, run the **display rip process-id** command to check BFD for RIP configurations on the specified interface. **interface** [*interface-type interface-number*] **verbose**

```
<Huawei> display rip 1 interface ethernet1/0/0 verbose
Ethernet1/0/0 (81.1.1.1)
  State      : UP           MTU : 500
  Metricin   : 0
  Metricout  : 1
  Input      : Enabled     Output      : Enabled
  Protocol   : RIPv1 Compatible (Non-Standard)
  Send       : RIPv1 Packets
  Receive    : RIPv1 Packets, RIPv2 Multicast and Broadcast Packets
  Poison-reverse : Disabled
  Split-Horizon : Enabled
  Authentication type : None
  Replay Protection : Disabled
  BFD        : Enabled (Static)
  Summary Address (es):
    1.1.0.0/16
```

3.12 Configuring the Network Management Function in RIP

By binding RIP to MIBs, you can view and configure RIP through the NMS.

3.12.1 Establishing the Configuration Task

Before binding RIP to MIBs, familiarize yourself with the applicable environment, complete the pre-configuration tasks, and obtain the data required for the configuration. This will help you complete the configuration task quickly and accurately.

Applicable Environment

After performing configuration procedures in this section, you can bind RIP to a MIB.

Pre-configuration Tasks

Before configuring the network management function in RIP, complete the following tasks:

- Configuring IP addresses for interfaces to ensure that neighboring nodes are reachable at the network layer
- [Configuring Basic RIP Functions](#)

Data Preparation

None.

3.12.2 Binding RIP to MIBs

Before binding RIP to MIBs, you need to specify the RIP process ID.

Context

Do as follows on the RIP router:

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
rip mib-binding process-id
```

RIP is bound to MIBs.

This command is used to bind a RIP process ID to MIBs and specify the ID of the RIP process that accepts Simple Network Management Protocol (SNMP) requests.

---End

3.12.3 Checking the Configuration

After RIP and MIBs are successfully bound, you can view binding information in the current RIP configuration.

Prerequisites

The configurations of the network management function in RIP are complete.

Procedure

Step 1 Run the **display current-configuration** command to check the parameters that take effect on the router.

---End

3.13 Maintaining RIP

This section describes how to reset RIP connections and clear RIP information.

3.13.1 Resetting RIP

Restarting RIP can reset RIP.

Context



CAUTION

The RIP neighbor relationship is deleted after you reset RIP connections with the **reset rip** command. Exercise caution when running this command.

To reset RIP connections, run the following **reset** commands in the user view.

Procedure

- Run the **reset rip *process-id* configuration** command in the user view to reset the parameters of the specified RIP process. When the RIP process starts, all parameters use default values.

----End

3.13.2 Clearing RIP

This section describes how to clear statistics about RIP counters.

Context



CAUTION

RIP information cannot be restored after it is cleared. Exercise caution when running the commands.

To clear RIP information, run the following **reset** command in the user view.

Procedure

- Run the **reset rip *process-id* statistics [interface { all | *interface-type* *interface-number* [neighbor *neighbor-ip-address*] }]** command in the user view to clear statistics about the counter that is maintained by a specified RIP process.

----End

3.14 Configuration Examples

In actual networking, RIP versions and whether to import external routes will affect which routes can be learned.

3.14.1 Example for Configuring RIP Version

Before using RIP, you need to configure basic RIP functions and specify a RIP version. You can run commands to view the configuration results.

Networking Requirements

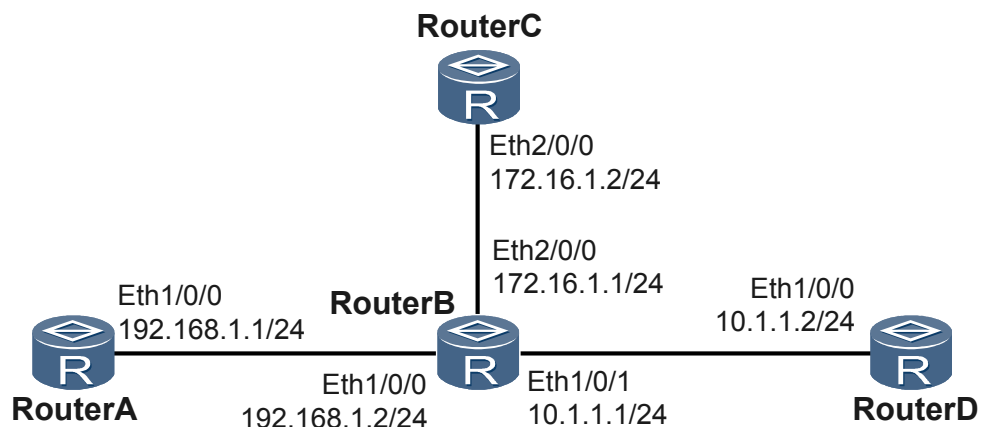
As shown in [Figure 3-1](#), it is required that RIP be enabled on all interfaces of Router A, Router B, Router C, and Router D and the routers interconnect with each other by using RIPv2.



NOTE

AR150/200 is RouterA, RouterC, or RouterD.

Figure 3-1 Networking diagram for configuring the RIP version number



Configuration Roadmap

The configuration roadmap is as follows:

1. Configure an IP address for each interface to ensure that neighboring nodes are reachable at the network layer.
2. Enable RIP on each router and configure basic RIP functions.
3. Configure RIPv2 on each router and check information about subnet masks.

Data Preparation

To complete the configuration, you need the following data:

- Network segment (192.168.1.0) to be enabled with RIP on Router A
- Network segments (192.168.1.0, 172.16.0.0, and 10.0.0.0) to be enabled with RIP on Router B
- Network segment (172.16.0.0) to be enabled with RIP on Router C
- Network segment (10.0.0.0) to be enabled with RIP on Router D
- RIPv2 on Router A, Router B, Router C, and Router D

Procedure

Step 1 Configure an IP address for each interface.

The configuration details are not described here.

Step 2 Configure basic RIP functions.

Configure Router A.

```
[RouterA] rip
[RouterA-rip-1] network 192.168.1.0
[RouterA-rip-1] quit
```

Configure Router B.

```
[RouterB] rip
```

```
[RouterB-rip-1] network 192.168.1.0
[RouterB-rip-1] network 172.16.0.0
[RouterB-rip-1] network 10.0.0.0
[RouterB-rip-1] quit
```

Configure Router C.

```
[RouterC] rip
[RouterC-rip-1] network 172.16.0.0
[RouterC-rip-1] quit
```

Configure Router D.

```
[RouterD] rip
[RouterD-rip-1] network 10.0.0.0
[RouterD-rip-1] quit
```

Check the RIP routing table of Router A.

```
[RouterA] display rip 1 route
Route Flags: R - RIP, T - TRIP
              P - Permanent, A - Aging, S - Suppressed, G - Garbage-collect
-----
Peer 192.168.1.2 on Ethernet1/0/0
  Destination/Mask      Nexthop      Cost  Tag   Flags  Sec
  10.0.0.0/8            192.168.1.2  1     0     RA     14
  172.16.0.0/16         192.168.1.2  1     0     RA     14
```

The preceding display shows that the routes advertised by RIPv1 carry natural masks.

Step 3 Configure the RIP version number.

Configure RIPv2 on Router A.

```
[RouterA] rip
[RouterA-rip-1] version 2
[RouterA-rip-1] quit
```

Configure RIPv2 on Router B.

```
[RouterB] rip
[RouterB-rip-1] version 2
[RouterB-rip-1] quit
```

Configure RIPv2 on Router C.

```
[RouterC] rip
[RouterC-rip-1] version 2
[RouterC-rip-1] quit
```

Configure RIPv2 on Router D.

```
[RouterD] rip
[RouterD-rip-1] version 2
[RouterD-rip-1] quit
```

Step 4 Verify the configuration.

Check the RIP routing table of Router A.

```
[RouterA] display rip 1 route
Route Flags: R - RIP
              A - Aging, S - Suppressed, G - Garbage-collect
-----
Peer 192.168.1.2 on Ethernet1/0/0
  Destination/Mask      Nexthop      Cost  Tag   Flags  Sec
  10.1.1.0/24           192.168.1.2  1     0     RA     32
  172.16.1.0/24         192.168.1.2  1     0     RA     32
```

The preceding display shows that the routes advertised by RIPv2 carry subnet masks.

---End

Configuration Files

- Configuration file of Router A

```
#
 sysname RouterA
#
interface Ethernet1/0/0
 ip address 192.168.1.1 255.255.255.0
#
rip 1
 version 2
 network 192.168.1.0
#
return
```

- Configuration file of Router B

```
#
 sysname RouterB
#
interface Ethernet1/0/0
 ip address 192.168.1.2 255.255.255.0
#
interface Ethernet1/0/1
 ip address 10.1.1.1 255.255.255.0
#
interface Ethernet2/0/0
 ip address 172.16.1.1 255.255.255.0
#
rip 1
 version 2
 network 192.168.1.0
 network 172.16.0.0
 network 10.0.0.0
#
return
```

- Configuration file of Router C

```
#
 sysname RouterC
#
interface Ethernet2/0/0
 ip address 172.16.1.2 255.255.255.0
#
rip 1
 version 2
 network 172.16.0.0
#
return
```

- Configuration file of Router D

```
#
 sysname RouterD
#
interface Ethernet1/0/0
 ip address 10.1.1.2 255.255.255.0
#
rip 1
 version 2
 network 10.0.0.0
#
return
```

4 RIPng Configuration

About This Chapter

RIPng is an extension of RIP for support of IPv6.

[4.1 RIPng Overview](#)

RIPng is a distance-vector routing protocol, which measures the distance to the destination host by the hop count.

[4.2 RIPng Features Supported by the AR150/200](#)

The RIPng features supported by the AR150/200 include split horizon and poison reverse.

[4.3 Configuring Basic RIPng Functions](#)

To implement RIPng features, you need to configure basic RIPng functions, including creating RIPng processes and enabling RIPng on interfaces.

[4.4 Configuring RIPng Route Attributes](#)

By setting RIPng route attributes, you can change RIPng routing policies.

[4.5 Controlling the Advertising of RIPng Routing Information](#)

To meet the requirements of complex networks, it is required to accurately control the advertising of RIPng routing information.

[4.6 Controlling the Receiving of RIPng Routing Information](#)

To meet the requirements of complex networks, it is required to accurately control the receiving of RIPng routing information.

[4.7 Optimizing a RIPng Network](#)

You can adjust and optimize the RIPng network performance by configuring RIPng timers, split horizon, poison reverse, and zero field check.

[4.8 Maintaining RIPng](#)

This section describes how to clear statistics of a specified RIPng process.

[4.9 Configuration Examples](#)

In actual networking, different RIPng features have different applications.

4.1 RIPng Overview

RIPng is a distance-vector routing protocol, which measures the distance to the destination host by the hop count.

The Routing Information Protocol Next Generation (RIPng) protocol is an extension of RIPv2 that is applied to IPv4 networks. Most RIP-related concepts are applicable to RIPng.

Extension of RIP

For IPv6 applications, RIPng extends RIP as follows:

- UDP port number: In RIPng, UDP port number 521 is used to send and receive routing information.
- Multicast group address: In RIPng, FF02::9 is used as the multicast group address of RIPng routers.
- Prefix length: In RIPng, the prefix length of a destination address is 128 bits (the mask length).
- Next-hop address: In RIPng, a next-hop address is a 128-bit IPv6 address.
- Source address: In RIPng, link-local address is used as the source address to send RIPng Update packets.

Operation Principle of RIPng

RIPng is a distance-vector routing protocol. It exchanges routing information by using User Datagram Protocol (UDP) packets through the port 521.

RIPng employs the hop count to measure the distance to the destination. The distance is called the routing metric. In RIPng, the hop count from the router to its directly connected network is 0, and the hop count from the router to a network, which can be reached through another router, is 1. The hop count that is equal to or exceeds 16 is defined as infinity, indicating that the destination network or host is unreachable.

By default, RIPng sends an Update packet every 30 seconds. If no Update packet is received from a neighbor in 180 seconds, RIPng marks all the routes learned from the neighbor as unreachable. If no Update packet is received from a neighbor in 300 seconds, RIPng deletes the routes of the neighbor from the routing table.

To prevent routing loops, RIPng supports split horizon and poison reverse. In addition, RIPng can import routes from other routing protocols.

Each router running RIPng manages a routing database, which contains routing entries to all accessible destinations on a network. These routing entries contain the following information:

- Destination address: indicates the IPv6 address of a host or network.
- Next-hop address: indicates the address of the next router to the destination.
- Interface: indicates the interface through which an IP packet is forwarded.
- Cost: indicates the hop count to the destination. The value is an integer that ranges from 0 to 16. If the value is 16, it indicates that the destination host or network is unreachable.
- Timer: indicates the time since a routing entry is last updated. The timer is reset to 0 when a routing entry is updated.

- Route tag: indicates a label that differentiates routes of interior routing protocols and those of exterior routing protocols.

4.2 RIPng Features Supported by the AR150/200

The RIPng features supported by the AR150/200 include split horizon and poison reverse.

In the AR150/200, you can modify the routing policy of RIPng by configuring RIPng route attributes. You can also control the advertising and receiving of RIPng routing information to meet the requirements of a complex network. On certain networks, you can configure RIPng features to optimize the RIPng network performance.

NOTE

The RIPng function is used with a license. To use the RIPng function, apply for and purchase the following license from the Huawei local office:

- AR150&200 Value-Added Data Package

4.3 Configuring Basic RIPng Functions

To implement RIPng features, you need to configure basic RIPng functions, including creating RIPng processes and enabling RIPng on interfaces.

4.3.1 Establishing the Configuration Task

To make a Router learn the routes to the network segment of an interface, ensure that the link status of the interface is Up.

Applicable Environment

The configuration of basic RIPng functions involves the configuration of basic RIPng features. After the configuration, the RIPng features are available.

During the RIPng configuration, you must enable RIPng in the system view first. If you run RIPng-related commands in the interface view, these commands take effect only after RIPng is enabled in the system view.

Pre-configuration Tasks

Before configuring basic RIPng functions, complete the following tasks:

- Enabling IPv6 on the router
- Configuring IPv6 addresses for interfaces to ensure that neighboring nodes are reachable at the network layer

Data Preparation

To configure basic RIPng functions, you need the following data.

No.	Data
1	RIPng process ID

No.	Data
2	Interface to be enabled with RIPng

4.3.2 Enabling RIPng and Entering the RIPng View

Creating RIPng processes is the prerequisite to performing RIPng configurations. When creating RIPng processes, you can also enter the RIPng view to perform configurations.

Context

Do as follows on the router to be enabled with RIPng:

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
ripng [ process-id ]
```

The RIPng process is enabled and the RIPng view is displayed.

When only one RIPng process runs, *process-id* does not need to be specified. That is, *process-id* defaults to 1.

After the RIPng process is cancelled, the **ripng process-id enable** command needs to be reconfigured on an interface.

---End

4.3.3 Enabling RIPng in the Interface View

After an interface is associated with a RIPng process, routing information on this interface can be exchanged through RIPng.

Context

Do as follows on the router to be enabled with RIPng:

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
interface interface-type interface-number
```

The interface view is displayed.

The interface is at the network side of the router. That is, the router is connected to other devices through this interface. To enable the router to learn routes to the network segment where the interface resides, ensure that the link status of the interface is Up.

Step 3 Run:

```
ripng process-id enable
```

RIPng is enabled on the specified interface.

 **NOTE**

In the interface view, this command cannot be executed if IPv6 is not enabled.

This command is inapplicable to ATM interfaces.

If the router connects to other devices through multiple interfaces, repeatedly perform Step 2 and Step 3.

----End

4.3.4 Checking the Configuration

After basic RIPng functions are successfully configured, you can view the configuration and routing information of RIPng.

Prerequisites

The configurations of basic RIPng functions are complete.

Procedure

- Run the **display ripng [process-id]** command to check the configuration of the RIPng process.
- Run the **display ripng process-id route** command to check all the RIPng routes that are learned from other routers.
- Run the **display default-parameter ripng** command to check the default RIPng configuration.
- Run the **display ripng process-id statistics interface { all | interface-type interface-number [verbose | neighbor neighbor-ipv6-address] }** command to check statistics about RIPng interfaces.

----End

4.4 Configuring RIPng Route Attributes

By setting RIPng route attributes, you can change RIPng routing policies.

4.4.1 Establishing the Configuration Task

RIPng route attributes include the RIPng preference and interface metric.

Applicable Environment

To meet the requirements of a complex network, you can change RIPng routing policies by configuring RIPng route attributes. After performing configuration procedures in this section, you can:

- Affect route selection by changing the additional metric of a RIPng interface.
- Change the matching order of routing protocols by configuring the RIPng preference when multiple routing protocols discover routes to the same destination.
- Implement load balancing among multiple equal-cost routes.

Pre-configuration Tasks

Before configuring RIPng route attributes, complete the following tasks:

- Configuring IPv6 addresses for interfaces to ensure that neighboring nodes are reachable at the network layer
- [3.3 Configuring Basic RIP Functions](#)

Data Preparation

To configure RIPng route attributes, you need the following data.

No.	Data
1	Additional metric of the interface
2	RIPng preference
3	Maximum number of equal-cost routes

4.4.2 Configuring the RIPng Preference

When there are routes discovered by multiple routing protocols on the same router, you can make the router prefer RIPng routes by setting the RIPng preference.

Context

Each routing protocol has its preference, according to which a routing policy selects the optimal route. The RIPng preference can be set manually. The greater the value is, the lower the preference is.

Do as follows on the RIPng router:

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
ripng [ process-id ]
```

The RIPng process is enabled and the RIPng view is displayed.

Step 3 Run:

```
preference { preference | route-policy route-policy-name } *
```

The RIPng preference is set.

----End

4.4.3 Configuring Additional Metrics of an Interface

You can set additional metrics for received and sent RIPng routes by using different commands.

Context

The additional route metric is the metric (hop count) to be added to the original metric of a RIPng route.

- The **ripng metricin** command is used to configure a device to add an additional metric to a received route before the device adds the route to its routing table, causing the metric of the route in the routing table to change. Running this command affects route selection on the device and other devices.
- The **ripng metricout** command is used to configure a device to add an additional metric to a route before the device advertises the route, keeping the metric of the route in the routing table unchanged. Running this command does not affect route selection on the local device but will affect route selection of other devices.

You can specify the value of the metric to be added to the RIPng route that passes the filtering policy by specifying *value1* through an IPv6 prefix list. If a RIPng route does not pass the filtering, its metric is increased by 1.

Do as follows on the RIPng router:

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
interface interface-type interface-number
```

The interface view is displayed.

Step 3 Run:

```
ripng metricin value
```

The metric added to a received route is set.

Step 4 Run:

```
ripng metricout { value | ipv6-prefix ipv6-prefix-name value1 }
```

The metric added to a sent route is set.

 **NOTE**

If the router connects to other RIPng routers through multiple interfaces, repeatedly perform Step 2 to Step 4 until metrics of all links are set.

----End

4.4.4 Configuring the Maximum Number of Equal-Cost Routes

By setting the maximum number of equal-cost RIPng routes, you can change the number of routes for load balancing.

Context

Do as follows on the RIPng router:

Procedure

- Step 1** Run:
- ```
system-view
```
- The system view is displayed.
- Step 2** Run:
- ```
ripng [ process-id ]
```
- The RIPng view is displayed.
- Step 3** Run:
- ```
maximum load-balancing number
```
- The maximum number of equal-cost routes is set.
- End

## 4.4.5 Checking the Configuration

After RIPng route attributes are successfully set, you can view the configuration and routing information of RIPng.

### Prerequisites

The configurations of RIPng route attributes are complete.

### Procedure

- Run the **display ripng [ process-id ]** command to check the running status and configuration of RIPng.
  - Run the **display ripng process-id database** command to check all activated routes in the RIPng database.
  - Run the **display ripng process-id route** command to check all the RIPng routes that are learned from other routers.
- End

## 4.5 Controlling the Advertising of RIPng Routing Information

To meet the requirements of complex networks, it is required to accurately control the advertising of RIPng routing information.

## 4.5.1 Establishing the Configuration Task

RIPng routing information can be advertised through route summarization, default routes, and imported external routes.

### Applicable Environment

To meet the requirements of a complex network, you need to control the advertising of RIPng routing information accurately. After performing configuration procedures in this section, you can:

- Advertise default routes to neighbors.
- Suppress interfaces from sending RIPng Update packets.
- Import external routes from various routing protocols and filter routes to be advertised.

### Pre-configuration Tasks

Before configuring the router to control the advertising of RIPng routing information, complete the following tasks:

- Configuring IPv6 addresses for interfaces to ensure that neighboring nodes are reachable at the network layer
- [3.3 Configuring Basic RIP Functions](#)

### Data Preparation

To control the advertising of RIPng routing information, you need the following data.

| No. | Data                                                              |
|-----|-------------------------------------------------------------------|
| 1   | Metric of the default route to be advertised                      |
| 2   | Protocol name and process ID of the external route to be imported |

## 4.5.2 Configuring RIPng Route Summarization

By configuring a RIPng router to advertise the summarized IPv6 address on an interface, you can save the space used by RIPng routes in the routing table. You can also set parameters to prevent an interface from learning the same summarized route.

### Context

This configuration is to configure the RIPng router to advertise the summarized IPv6 prefix rather than specific routes on an interface.

Do as follows on the RIPng router:

### Procedure

- Step 1** Run:
- ```
system-view
```

The system view is displayed.

Step 2 Run:

```
interface interface-type interface-number
```

The interface view is displayed.

Step 3 Run:

```
ripng summary-address ipv6-address prefix-length [ avoid-feedback ]
```

RIPng route summarization is configured.

---End

4.5.3 Configuring RIPng to Advertise the Default Routes

There are two methods of advertising RIPng default routes. You can configure a router to advertise RIPng default routes according to the actual networking. Additionally, you can specify the cost of the default routes to be advertised.

Context

Do as follows on the RIPng router:

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
interface interface-type interface-number
```

The interface view is displayed.

Step 3 Run:

```
ripng default-route { only | originate } [ cost cost ]
```

RIPng is configured to advertise a default route.

You can configure RIPng to advertise default routes as required:

- **only**: advertises only IPv6 default routes (::/0) and suppresses the advertising of other routes.
- **originate**: advertises IPv6 default routes (::/0) and does not affect the advertising of other routes.

A RIPng default route is forcibly advertised by using an Update packet through a specified interface, regardless of whether this route exists in the IPv6 routing table.

---End

4.5.4 Configuring the Default Cost for External Routes Imported by RIPng

If RIPng imports routes from other routing protocols, but no metric is specified, you can set the default metric for imported external routes.

Context

Do as follows on the RIPng router:

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
ripng [ process-id ]
```

The RIPng view is displayed.

Step 3 Run:

```
default-cost cost
```

The default cost is set for the external routes imported by RIPng.

If no metric is specified, this command can be used to set the default cost for the external routes imported by RIPng from other routing protocols.

----End

4.5.5 Configuring RIPng to Import External Routes

Similar to RIP, RIPng can import external routes to enrich routing information.

Context

Do as follows on the RIPng router:

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
ripng [ process-id ]
```

The RIPng view is displayed.

Step 3 (Optional) Run:

```
default-cost cost
```

The default cost is set for imported external routes.

Step 4 Run:

```
import-route protocol [ process-id ] [ cost cost | route-policy route-policy-name ]  
*
```

External routes are imported.

If no cost is specified for imported routes, the default cost is used.

Step 5 (Optional) Run:

```
filter-policy ipv6-prefix ipv6-prefix-name export [ protocol [ process-id ] ]
```

RIPng is configured to filter the imported routing information.

RIPng can filter the imported routes based on an IPv6 prefix list. Only the routes that meet the match conditions are advertised to neighbors. If *protocol* is not specified in the command, all the routing information to be advertised will be filtered, including the imported routes and local RIPng routes (directly connected routes).

----End

4.5.6 Checking the Configuration

After the function of controlling the advertising of RIPng routing information is successfully configured, you can view RIPng routing information.

Prerequisites

The configurations of controlling the advertising of RIPng routing information are complete.

Procedure

- Run the **display ripng process-id database** command to check all activated routes in the RIPng database.
- Run the **display ripng process-id route** command to check all the RIPng routes that are learned from other routers.

----End

4.6 Controlling the Receiving of RIPng Routing Information

To meet the requirements of complex networks, it is required to accurately control the receiving of RIPng routing information.

4.6.1 Establishing the Configuration Task

Before controlling the receiving of RIPng routes, familiarize yourself with the applicable environment, complete the pre-configuration tasks, and obtain the data required for the configuration. This will help you complete the configuration task quickly and accurately.

Applicable Environment

To meet the requirements of a complicated networking environment, you need to control the receiving of RIPng routing information accurately. After performing configuration procedures in this section, you can:

- Disable an interface from receiving RIPng Update packets.
- Filter the received routing information.
- Import external routes from various routing protocols and filter the imported routes.

Pre-configuration Tasks

Before configuring the router to control the receiving of RIPng routing information, complete the following tasks:

- Configuring IPv6 addresses for interfaces to ensure that neighboring nodes are reachable at the network layer
- **Configuring Basic RIPng Functions**

Data Preparation

To control the receiving of RIPng routing information, you need the following data.

No.	Data
1	Name of the IP prefix list used to filter routes

4.6.2 Configuring RIPng to Filter the Received Routes

By configure an IPv6 prefix list to filter received routes, you can configure a router to selectively receive routes.

Context

Do as follows on the RIPng router:

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
ripng [ process-id ]
```

The RIPng view is displayed.

Step 3 Run:

```
filter-policy ipv6-prefix ipv6-prefix-name import
```

The imported routes are filtered.

You can specify an IPv6 prefix list to filter the imported routes. Only the routes that pass the filtering can be added to the RIPng routing table.

---End

4.6.3 Checking the Configuration

After the function of controlling the receiving of RIPng routing information is successfully configured, you can view RIPng routing information.

Prerequisites

The configurations of controlling the receiving of RIPng routing information are complete.

Procedure

- Run the **display ripng process-id database** command to check all activated routes in the RIPng database.
- Run the **display ripng process-id route** command to check all the RIPng routes that are learned from other routers.

---End

4.7 Optimizing a RIPng Network

You can adjust and optimize the RIPng network performance by configuring RIPng timers, split horizon, poison reverse, and zero field check.

4.7.1 Establishing the Configuration Task

Before adjusting and optimizing the RIPng network performance, familiarize yourself with the applicable environment, complete the pre-configuration tasks, and obtain the data required for the configuration. This will help you complete the configuration task quickly and accurately.

Applicable Environment

On certain networks, you need to configure RIPng features and optimize the performance of a RIPng network. After performing configuration procedures in this section, you can:

- Change the convergence speed of the RIPng network by adjusting RIPng timers.
- Configure split horizon and poison reverse to prevent routing loops.

Pre-configuration Tasks

Before optimizing a RIPng network, complete the following tasks:

- Configuring IPv6 addresses for interfaces to ensure that neighboring nodes are reachable at the network layer
- [Configuring Basic RIPng Functions](#)

Data Preparation

To optimize a RIPng network, you need the following data.

No.	Data
1	Values of timers

4.7.2 Configuring RIPng Timers

RIPng has three timers: Update timer, Age timer and Garbage-collect timer. If the three RIPng timers are configured improperly, routes become unstable.

Context

 **NOTE**

Route flapping occurs if the values of the four RIPng timers are set improperly. The relationship between the values is as follows: $update < age$, $update < garbage-collect$. For example, if the update time is longer than the aging time, and a RIPng route changes within the update time, the router cannot inform its neighbors of the change on time.

By default, the Update timer is 30s; the Age timer is 180s; the Garbage-collect timer is 120s.

Do as follows on the RIPng router:

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
ripng [ process-id ]
```

The RIPng view is displayed.

Step 3 Run:

```
timers ripng update age garbage-collect
```

RIPng timers are configured.

---End

4.7.3 Setting the Interval for Sending Update Packets and the Maximum Number of Packets Sent Each Time

By setting the interval for sending packets and the maximum number of packets to be sent each time, you can optimize the RIPng performance.

Context

Do as follows on the RIPng router:

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
interface interface-type interface-number
```

The interface view is displayed.

Step 3 Run:

```
ripng pkt-transmit { interval interval | number pkt-count }*
```

The interval for sending RIPng Update packets and the maximum number of packets sent each time are set on the specified interface.

----End

4.7.4 Configuring Split Horizon and Poison Reverse

You can configure split horizon and poison reverse to prevent routing loops.

Context

Split horizon is a method of preventing routing loops by preventing the router from advertising a route back onto the interface from which the route is learned. On NBMA networks such as FR networks and X.25 networks, split horizon is disabled by default.

Poison reverse is another method of preventing routing loops by enabling the router to advertise a route as unreachable back through the interface from which the route is learned.

If both split horizon and poison reverse are configured, only poison reverse takes effect.

Do as follows on the RIPng router:

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
interface interface-type vlan-id
```

The interface view is displayed.

Step 3 Run the following command as required:

● Run:

```
ripng split-horizon
```

Split horizon is enabled.

● Run:

```
ripng poison-reverse
```

Poison reverse is enabled.

----End

4.7.5 Enabling the Zero Field Check for RIPng Packets

In a RIPng packet, there are certain fields whose values must be 0. These fields are called zero fields. If the values of these zero fields in some RIPng packets are not 0s, these RIPng packets are ignored.

Context

Do as follows on the RIPng router:

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
ripng [ process-id ]
```

The RIPng view is displayed.

Step 3 Run:

```
checkzero
```

The zero field check is configured for RIPng packets.

---End

4.7.6 Checking the Configuration

After the function of adjusting and optimizing the RIPng network performance is successfully configured, you can view routing information, neighbor information, and interface information of RIPng.

Prerequisites

The configurations of adjusting and optimizing the RIPng network performance are complete.

Procedure

- Run the **display ripng [process-id]** command to check the configuration of the RIPng process.
- Run the **display ripng process-id database [verbose]** command to check all activated routes in the RIPng database.
- Run the **display ripng process-id interface [interface-type interface-number] [verbose]** command to check information about the RIPng interface.
- Run the **display ripng process-id neighbor [verbose]** command to check information about RIPng neighbors.
- Run the **display ripng process-id route** command to check all the RIPng routes that are learned from other routers.

---End

4.8 Maintaining RIPng

This section describes how to clear statistics of a specified RIPng process.

4.8.1 Clearing RIPng

This section describes how to clear statistics about RIPng counters.

Context



CAUTION

RIPng information cannot be restored after it is cleared. Exercise caution when running the commands.

To clear RIPng information, run the following **reset** command in the user view.

Procedure

- Run the **reset ripng process-id statistics [interface { all | interface-type interface-number [neighbor neighbor-ip-address] }** command in the user view to clear statistics about the counter that is maintained by a specified RIPng process.

----End

4.9 Configuration Examples

In actual networking, different RIPng features have different applications.

4.9.1 Example for Configuring Basic RIPng Functions

Before using RIP, you need to configure basic RIP functions.

Networking requirements

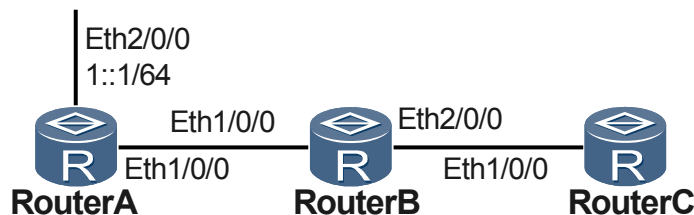
As shown in [Figure 4-1](#), the prefix length of all the IPv6 addresses is 64 bits and the neighboring routers are assigned IPv6 link-local addresses. All the routers must learn IPv6 routing information using RIPng.



NOTE

Only RouterC can use the AR150/200.

Figure 4-1 Networking diagram of configuring basic RIPng functions



Configuration Roadmap

The configuration roadmap is as follows:

1. Configure IPv6 addresses for interfaces.
2. Enable RIPng on each router so that the routers can communicate with each other.

Data Preparation

To complete the configuration, you need the following data:

- RIPng1 enabled on each router

Procedure

Step 1 Configure an IPv6 address for each interface. The configuration details are not mentioned here.

Step 2 Configure basic RIPng functions.

Configure RouterA.

```
[RouterA] ripng 1
[RouterA-ripng-1] quit
[RouterA] interface ethernet 2/0/0
[RouterA-Ethernet2/0/0] ripng 1 enable
[RouterA-Ethernet2/0/0] quit
[RouterA] interface ethernet 1/0/0
[RouterA-Ethernet1/0/0] ripng 1 enable
[RouterA-Ethernet1/0/0] quit
```

Configure RouterB.

```
[RouterB] ripng 1
[RouterB-ripng-1] quit
[RouterB] interface ethernet 1/0/0
[RouterB-Ethernet1/0/0] ripng 1 enable
[RouterB-Ethernet1/0/0] quit
[RouterB] interface ethernet 2/0/0
[RouterB-Ethernet2/0/0] ripng 1 enable
[RouterB-Ethernet2/0/0] quit
```

Configure RouterC.

```
[RouterC] ripng 1
[RouterC-ripng-1] quit
[RouterC] interface ethernet 1/0/0
[RouterC-Ethernet1/0/0] ripng 1 enable
[RouterC-Ethernet1/0/0] quit
```

View the RIPng routing table of RouterB.

```
[RouterB] display ripng 1 route
Route Flags: A - Aging, G - Garbage-collect
-----
Peer FE80::A19:A6FF:FECE:7D4C on Ethernet1/0/0
Dest 1::/64,
    via FE80::A19:A6FF:FECE:7D4C, cost 1, tag 0, A, 25 Sec
```

View the RIPng routing table of RouterC.

```
[RouterC] display ripng 1 route
Route Flags: A - Aging, G - Garbage-collect
-----
Peer FE80::2E0:FCFF:FE01:9 on Ethernet1/0/0
```

```
Dest 1::/64,  
  via FE80::2E0:FCFF:FE01:9, cost 2, tag 0, A, 4 Sec
```

----End

Configuration Files

- Configuration file of RouterA

```
#  
 sysname RouterA  
#  
 ipv6  
#  
 interface Ethernet1/0/0  
  ipv6 enable  
  ipv6 address auto link-local  
  ripng 1 enable  
#  
 interface Ethernet2/0/0  
  ipv6 enable  
  ipv6 address 1::1/64  
  ripng 1 enable  
#  
 ripng 1  
#  
 return
```

- Configuration file of RouterB

```
#  
 sysname RouterB  
#  
 ipv6  
#  
 interface Ethernet1/0/0  
  ipv6 enable  
  ipv6 address auto link-local  
  ripng 1 enable  
#  
 interface Ethernet2/0/0  
  ipv6 enable  
  ipv6 address auto link-local  
  ripng 1 enable  
#  
 ripng 1  
#  
 return
```

- Configuration file of RouterC

```
#  
 sysname RouterC  
#  
 ipv6  
#  
 interface Ethernet1/0/0  
  ipv6 enable  
  ipv6 address auto link-local  
  ripng 1 enable  
#  
 ripng 1  
#  
 return
```

5 OSPF Configuration

About This Chapter

OSPF, which is developed by the IETF, is a link-state IGP. OSPF is widely used in access networks and MANs.

[5.1 OSPF Overview](#)

OSPF is a link-state IGP. At present, OSPFv2 is intended for IPv4.

[5.2 OSPF Features Supported by the AR150/200](#)

The AR150/200 supports various OSPF features, including multi-process, authentication, Smart-discover, GR, VPN multi-instance, sham link, BFD, OSPF-BGP association, and GTSM.

[5.3 Configuring Basic OSPF Functions](#)

This section describes how to configure basic OSPF functions.

[5.4 Configuring OSPF on the NBMA or P2MP Network](#)

This section describes how to configure OSPF and modify attributes on the NBMA or point-to-multipoint (P2MP) network to flexibly construct the OSPF network.

[5.5 Configuring an OSPF Route Selection Rule](#)

You can configure an OSPF route selection rule to meet requirements of complex networks.

[5.6 Controlling OSPF Routing Information](#)

You can control the advertising and receiving of OSPF routing information and import routes of other protocols.

[5.7 Configuring an OSPF Stub Area](#)

Configuring a non-backbone area as a stub area can reduce routing entries in the area in an AS does not transmit routes learned from other areas in the AS or AS external routes. This reduces bandwidth and storage resource consumption.

[5.8 Configuring an NSSA](#)

Configuring a non-backbone area on the border of an AS as an NSSA does not transmit routes learned from other areas in the AS but imports AS external routes. This reduces bandwidth and storage resource consumption on the router.

[5.9 Configuring BFD for OSPF](#)

After BFD for OSPF is enabled, when a link fails, the router rapidly detects the failure, notifies the OSPF process or interface of the fault, and instructs OSPF to recalculate routes. This speeds up OSPF network convergence.

[5.10 Configuring OSPF GR](#)

Configuring OSPF GR to avoid traffic interruption and route flapping caused by the active/standby switchover.

[5.11 Improving Security of an OSPF Network](#)

On a network demanding high security, you can adopt the GTSM mechanism and configure OSPF authentication to improve the security of the OSPF network.

[5.12 Configuring the Network Management Function of OSPF](#)

OSPF supports the network management function. You can bind the OSPF MIB to a certain OSPF process, and configure the trap function and log function.

[5.13 Maintaining OSPF](#)

Maintaining OSPF involves resetting OSPF and clearing OSPF statistics.

[5.14 Configuration Examples](#)

This section provides several configuration examples of OSPF together with the configuration flowchart. The configuration examples explain networking requirements, configuration notes, and configuration roadmap.

5.1 OSPF Overview

OSPF is a link-state IGP. At present, OSPFv2 is intended for IPv4.

Defined by the Internet Engineering Task Force (IETF), the Open Shortest Path First (OSPF) protocol is an Interior Gateway Protocol (IGP) implemented on the basis of the link status.

NOTE

In this chapter, OSPF refers to OSPFv2, unless otherwise specified.

OSPF Features

OSPF has the following features:

- Wide applications
OSPF is applicable to networks of various sizes and even to the network consisting of hundreds of routers.
- Fast convergence
Once the network topology changes, Update packets are transmitted to synchronize the link state databases (LSDBs) of all the routers within the Autonomous System (AS).
- Loop-free
According to the collected link status, OSPF calculates routes with the shortest path tree algorithm. This algorithm ensures the generation of loop-free routes.
- Area division
An AS can be divided into different areas to facilitate AS management. After the area partition, an LSDB stores routing information only of the local area. The reduce of LSDB size dramatically reduces memory and CPU usage. In addition, less bandwidth is consumed because of the decrease in routing information transmitted within the AS.
- Equal-cost routes
OSPF supports multiple equal-cost routes to the same destination.
- Routing hierarchy
Four types of routing are available. They are listed in the descending order of priority: intra-area routes, inter-area routes, Type 1 external routes, and Type 2 external routes.
- Authentication
Area-based and interface-based packet authentication guarantees the security of packet interaction.
- Multicast
Multicast packets are transmitted only on certain types of links to reduce the interference for some devices.

Process of OSPF Route Calculation

The process of calculating OSPF routes is as follows:

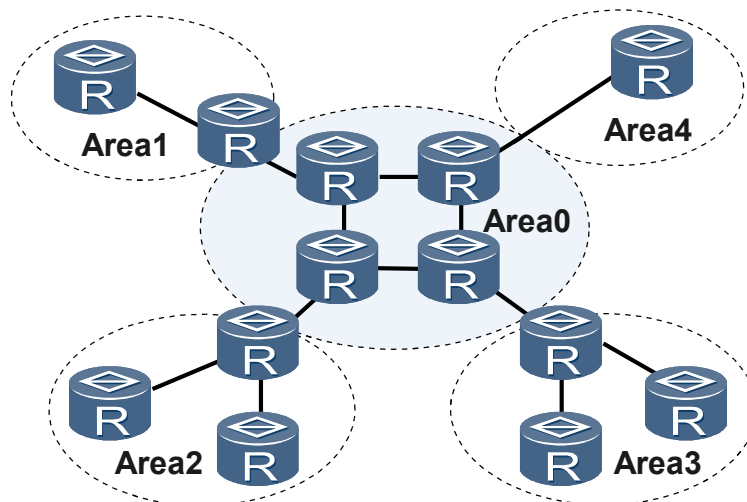
1. Based on the surrounding network topology, each OSPF device originates a Link State Advertisement (LSA). The router then transmits Update packets containing the LSAs to other OSPF devices.

2. Each OSPF device collects the LSAs from other devices, and all these LSAs compose the LSDB. An LSA describes the network topology around a router, whereas an LSDB describes the network topology of the whole AS.
3. OSPF devices transform the LSDB into a weighted directed map. The weighted directed map reflects the topology of the entire network. All routers in the same area have the same map.
4. According to the directed map, each router uses the Shortest Path First (SPF) algorithm to calculate the shortest path tree, regarding itself as the root. The tree displays the routes to each node in the AS.

Area Division

The number of routers increases with the unceasing expansion of the network scale. This leads to a large LSDB on each router. As a result, the load of each router is very heavy. OSPF solves this problem by dividing an AS into different areas. An area is regarded as a device group logically. Each group is identified by an area ID. On the border of an area resides a router rather than a link. A network segment (or a link) belongs to only one area. That is, the area to which each OSPF interface belongs must be specified, as shown in [Figure 5-1](#).

Figure 5-1 OSPF area division



After area division, route aggregation can be performed on border routers to reduce the number of LSAs advertised to other areas. Route aggregation also minimizes the influence caused by changes in the topology.

Router Type

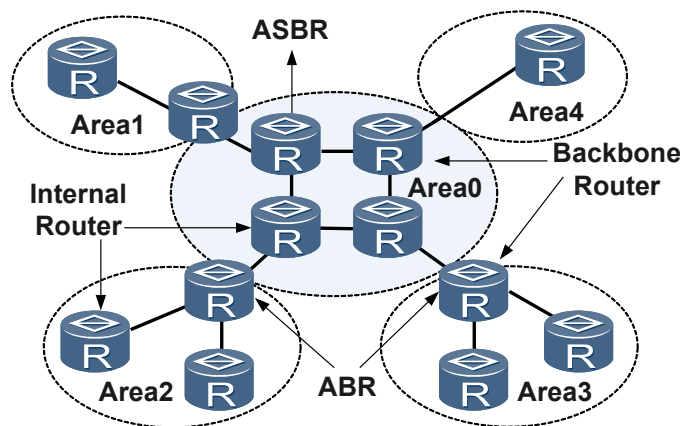
OSPF routers are classified into the following types according to their locations in the AS:

- Internal routers
All interfaces of the routers of this type belong to the same OSPF area.
- Area border routers (ABRs)

The routers of this type can belong to more than two areas, but one of the areas must be a backbone area. An ABR is used to connect the backbone area to the non-backbone areas. An ABR can be physically or logically connected to the backbone area.

- **Backbone routers**
 A minimum of one interface on the router of this type belongs to the backbone area. Therefore, all ABRs and the internal nodes in Area 0 are backbone routers.
- **AS boundary routers (ASBRs)**
 The router that exchanges routing information with other ASs is called an ASBR. The ASBR may not be located on the boundary of an AS. It can be an internal router or an ABR. When an OSPF device imports the external routing information, the device becomes an ASBR.

Figure 5-2 Types of OSPF routers



OSPF Network Types

OSPF classifies networks into four types according to the link layer protocol:

- **Broadcast:** If the link layer protocol is Ethernet or FDDI, OSPF defaults the network type to broadcast. In this type of networks, the following situations occur.
 - Hello packets and packets from the Designated Router (DR) are sent in multicast mode (224.0.0.5: indicates the reserved IP multicast addresses for OSPF routers).
 - Link State Update (LSU) packets are sent to the DR in multicast mode (224.0.0.6: indicates the reserved IP multicast address for the OSPF DR), and the DR forwards the LSU packets to destination 224.0.0.5.
 - Database Description (DD) packets, Link State Request (LSR) packets, and all retransmission packets are sent in unicast mode.
 - Link State Acknowledgement (LSAck) packets are usually sent in multicast mode (224.0.0.5). When a router receives repeated LSAs, or the LSAs are deleted due to the timeout of the maximum lifetime, LSAck packets are sent in unicast mode.
- **Non-Broadcast Multi-Access (NBMA):** If the link layer protocol is Frame Relay, ATM, or X.25, OSPF defaults the network type to NBMA. In this type of networks, protocol packets, such as Hello packets, DD packets, LSR packets, LSU packets, and LSAck packet, are transmitted in unicast mode.

- Point-to-Multipoint (P2MP): A P2MP network must be forcibly changed from other network types. In this type of networks, Hello packets are transmitted in multicast mode (224.0.0.5); DD packets, LSR packets, LSU packets, and LSAck packets are transmitted in unicast mode.
- Point-to-Point (P2P): If the link layer protocol is PPP, HDLC, or LAPB, OSPF defaults the network type to P2P. In this type of networks, protocol packets, such as Hello packets, DD packets, LSR packets, LSU packets, and LSAck packets, are transmitted in multicast mode (224.0.0.5).

5.2 OSPF Features Supported by the AR150/200

The AR150/200 supports various OSPF features, including multi-process, authentication, Smart-discover, GR, VPN multi-instance, sham link, BFD, OSPF-BGP association, and GTSM.

Multi-process

OSPF supports multi-process. More than one OSPF process can run on the same router because processes are mutually independent. Route interaction between different OSPF processes is similar to the interaction between different routing protocols.

An interface of a router belongs to only a certain OSPF process.

A typical application of OSPF multi-process is to run OSPF between PEs and CEs in the VPN where OSPF is also adopted in the backbone network. On the PEs, the two OSPF processes are independent of each other.

Authentication

OSPF supports packet authentication. Only the OSPF packets that pass the authentication can be received. If the packets fail to pass the authentication, the neighbor relationship cannot be established.

The AR150/200 supports two authentication modes:

- Area authentication mode
- Interface authentication mode

If both modes are available, the latter is preferred.

Smart-discover

Generally, routers periodically send Hello packets through interfaces that run OSPF, routers set up and maintain the neighbor relationship, and elect the DR and the Backup Designated Router (BDR) on the multi-access network (broadcast or NBMA) by exchanging Hello packets. When establishing the neighbor relationship or electing the DR and the BDR on the multi-access network, interfaces can send Hello packets only when the Hello timer expires. This affects the speed for establishing the neighbor relationship and electing the DR and the BDR.

NOTE

- The interval for sending Hello packets on an interface depends on the interval for sending Hello packets set on the interface.
- The default value of the interval for sending Hello packets varies with the network type.

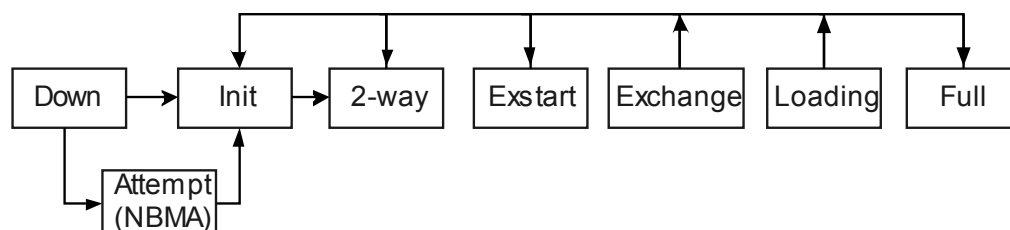
The Smart-discover function can solve the preceding problem.

- In broadcast and NBMA networks, the neighbor relationship can be established rapidly and a DR and a BDR on the networks can be elected rapidly.

When the neighbor status becomes 2-way for the first time, or it returns to Init from the 2-way or higher state as shown in **Figure 5-3**, the interface enabled with the Smart-discover function sends Hello packets to the neighbor without waiting for the timeout of the Hello timer when the interface finds that the status of the neighbor changes.

When the interface status of the DR and the BDR in the multi-access network changes, the interface enabled with the Smart-discover function sends Hello packets to the network segment and takes part in the DR or BDR election.

Figure 5-3 Changes of the neighbor state machine



- On P2P and P2MP networks, the adjacency relationship can be established rapidly. The principle is the same as that in broadcast and NBMA networks.

OSPF GR

When a router restarts or performs the active/standby switchover, it directly ages all routing entries in the Forward Information Base (FIB) table. This results in route interruption. In addition, neighboring routers remove this router from the neighbor list, and notify other routers. This causes the re-calculation of SPF. If this router recovers within a few seconds, the neighbor relationship becomes unstable. This results in route flapping.

After being enabled with OSPF Graceful Restart (GR), a router can ensure continuous packet forwarding if it restarts just for abnormalities. In such a case, route flapping is avoided during the short restart of the router.

NOTE

Unless otherwise specified, "protocol restart" in this document refers to restarting OSPF in GR mode.

When a router restarts OSPF, the GR Restarter does not age the forwarding information. At the same time, the GR Helper keeps the topology information or routes obtained from the GR Restarter for a period. This ensures that traffic forwarding is not interrupted when protocol restart occurs.

OSPF VPN Multi-instance

OSPF supports multi-instance, which can run between PEs and CEs in VPN networks.

In BGP MPLS VPN, many sites of one VPN can use OSPF as the internal routing protocol. The sites, however, are handled as being from different ASs. In this way, the OSPF routes learned on one site are transmitted as external routes to another site. This causes a heavy OSPF traffic and some avoidable network management problems.

In the AR150/200 implementation, you can configure domain IDs on a PE to differentiate the VPNs where different sites reside. Different sites in one VPN consider each other as if they were connected directly. Thus, PEs exchange OSPF routing information as if they were directly connected through a leased line. This improves network management and enhances the validity of the OSPF application.

 **NOTE**

For detailed configuration of this feature, refer to the Huawei AR150&200 Series Enterprise Routers *Configuration Guide - VPN*.

OSPF Sham Links

OSPF sham links are unnumbered P2P links between two PEs over an MPLS VPN backbone network.

Generally, BGP extended community attributes carry routing information over the MPLS VPN backbone between BGP peers. OSPF running on the remote PE uses this information to generate Type3 summary LSAs from PE to CE. These routes are considered as inter-area routes.

If a router, however, connects to PEs in its own area and establishes an intra-area route (backdoor route) to a particular destination, the VPN traffic always traverses the backdoor route rather than the backbone route. This is because OSPF intra-area routes in the routing table have relatively higher priorities. To prevent this, an unnumbered P2P sham link is configured between the PEs. This provides an intra-area path with a lower cost to the PE.

 **NOTE**

For configurations of OSPF sham links, refer to the Huawei AR150&200 Series Enterprise Routers *Configuration Guide - VPN*.

BFD for OSPF

By default, in broadcast networks, the interval for OSPF to send Hello packets is 10 seconds; in NBMA networks, the interval for sending Hello packets is 30 seconds, and the period for advertising that the neighbor is Down is four times the interval for sending Hello packets. If the router does not receive the Hello packet from the neighbor before the neighboring router becomes invalid, it deletes the neighbor. That is, the router detects the neighbor faults in seconds. This leads to the loss of a large number of packets in a high-speed network.

To solve the preceding problem in the current detection mechanism, Bidirectional Forwarding Detection (BFD) is developed. BFD can implement detection at the millisecond level. Instead of replacing the Hello mechanism of OSPF, BFD works with OSPF to fast detect the adjacency fault. BFD is used to notify OSPF of recalculating routes. This can correctly guide the packet forwarding.

Routing Management (RM) module exchanges routing information with the BFD module. Through RM, OSPF notifies BFD of dynamically setting up or deleting BFD sessions. The Event message of BFD is delivered to OSPF through RM.

The process of establishing and deleting a BFD session is as follows:

- Process of establishing a BFD session: If BFD feature is globally configured, BFD is enabled on an interface or a process, and the status of the OSPF neighbor is Full, OSPF uses RM to notify the BFD module of establishing the BFD session and negotiate related parameters of BFD.
- Process of deleting a BFD session: When BFD detects a link fault, BFD generates a Down event and notifies the upper protocol of the fault through RM. OSPF then responds to the

event and immediately deletes the adjacency relationship on the link. At this time, the status of the neighbor is not Full. This does not meet the requirements of establishing a BFD session. OSPF then uses RM to notify the BFD module of deleting the BFD session.

OSPF supports dynamically establishing or deleting a BFD session on broadcast, P2P, P2MP, and NBMA links.

Configure BFD according to the actual network environment. If time parameters are set incorrectly, network flapping occurs.

OSPF-BGP

When a new router is connected to the network, or a router restarts, the network traffic may be lost during BGP convergence. This is because the IGP route convergence is quicker than the BGP route convergence.

If the backup link exists, OSPF-BGP linkage makes a router that restarts or a router that is connected to the network start the stub router timer during the OSPF-BGP linkage. During the set linkage period, the router acts as the stub router by increasing the metrics of the links in the LSA generated by the router to 65535. Other OSPF routers are notified of not using the stub router to forward data. This ensures that the router is not used as the spanned router. This avoids traffic loss during traffic switchback because route convergence speed is slower than that of OSPF.

GTSM

The Generalized TTL Security Mechanism (GTSM) refers to the generic TTL security protection mechanism. GTSM protects services of the upper layer over the IP layer by checking whether the TTL value in the IP header is in a pre-defined range. In applications, GTSM is designed to protect the TCP/IP-based control plane (like routing protocols) from CPU-utilization attacks, such as CPU overload attacks.

5.3 Configuring Basic OSPF Functions

This section describes how to configure basic OSPF functions.

5.3.1 Establishing the Configuration Task

Before configuring basic OSPF functions, enable OSPF, specify the OSPF process and area, and establish OSPF neighbor relationships.

Applicable Environment

When OSPF is configured on multiple routers in the same area, most configuration data, such as the timer, filter, and aggregation, must be planned uniformly in the area. Incorrect configurations may cause neighboring routers to fail to send messages to each other or even causing routing information congestion and self-loops.

The OSPF-relevant commands that are configured in the interface view take effect regardless of whether OSPF is enabled. After OSPF is disabled, the OSPF-relevant commands also exist on interfaces.

Pre-configuration Tasks

Before configuring basic OSPF functions, complete the following tasks:

- Configuring a link layer protocol
- Configuring IP addresses for interfaces to ensure that neighboring routers are reachable at the network layer

Data Preparation

To configure basic OSPF functions, you need the following data.

No.	Data
1	Router ID
2	OSPF process ID
3	VPN instance name (if OSPF multi-instance is configured)
4	ID of the area to which an interface belongs
5	IP address of the network segment where an interface resides

5.3.2 Enabling OSPF

Create an OSPF process and specify a router ID to enable OSPF. After enabling OSPF, specify an interface on which the OSPF protocol is running and the area to which the interface belongs. After that, routes can be discovered and calculated in the AS.

Context

Before running OSPF on the router, specify a router ID for the router. The router ID is a 32-bit unsigned integer, which identifies the router in the AS. To ensure OSPF stability, manually set the router ID of each router during network planning.

This causes the link state database (LSDB) to unexpectedly grow. OSPF resolves this problem by partitioning an AS into different areas. The area is regarded as a logical group and each group is identified by an area ID. At the border of an area resides the router instead of a link. A network segment (or a link) belongs to only one area. The area to which each OSPF interface belongs must be specified.

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
ospf [ process-id | router-id router-id | vpn-instance vpn-instance-name ] *
```

The OSPF process is started, and the OSPF view is displayed.

- *process-id* specifies the process ID and the *process-id* value is 1 by default. The AR150/200 supports OSPF multi-process. Processes can be classified by service type. The AR150/200s exchange packets regardless of process IDs. Packets can be exchanged between AR150/200s with different process IDs.
- Each router ID in an OSPF process must be unique. Otherwise, the OSPF neighbor relationship cannot be established and the routing information is incorrect. By default, the system automatically selects the IP address of an interface as the router ID. Ensure that each router ID is unique in the AS when manually configuring router IDs. The IP address of an interface on the AR150/200 is generally configured as the router ID of the AR150/200.
- If a VPN instance is specified, the OSPF process belongs to the VPN instance; if a VPN instance is not specified, the OSPF process belongs to a public network instance.

Step 3 Run:

```
area area-id
```

The OSPF area view is displayed.

The OSPF areas can be classified into a backbone area with the area ID of 0 and non-backbone areas. The backbone area is responsible for forwarding inter-area routing information. The routing information between the non-backbone areas must be forwarded through the backbone area.

Step 4 Run:

```
network ip-address wildcard-mask [ description text ]
```

The network segments are configured to belong to the area. **description** is used to configure a description for the specified OSPF network segment.

OSPF can run on an interface properly only when the following conditions are met:

- The mask length of the IP address of an interface is greater than or equal to that specified by the **network** command.
- The primary IP address of an interface is on the network segment specified by the **network** command.

By default, OSPF uses a 32-bit host route to advertise the IP address of a loopback interface. To advertise routes to the network segment of the loopback interface, configure the network type as NBMA or broadcast in the interface view. For details, see [Configuring Network Types for OSPF Interfaces](#).

---End

5.3.3 (Optional) Creating OSPF Virtual Links

This section describes how to create logical links between backbone areas to ensure the OSPF network connectivity.

Context

After OSPF areas are defined, OSPF route updates between non-backbone areas are transmitted through a backbone area. Therefore, OSPF requires that all non-backbone areas maintain the connectivity with the backbone area and the backbone areas in different OSPF areas maintain the connectivity with each other. In real world situations, this requirement may not be met because of some restrictions. To resolve this problem, you can configure OSPF virtual links.

Perform the following steps on the router running OSPF.

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
ospf [ process-id ]
```

The OSPF process view is displayed.

Step 3 Run:

```
area area-id
```

The OSPF area view is displayed.

Step 4 Run:

```
vlink-peer router-id [ smart-discover | hello hello-interval | retransmit  
retransmit-interval | trans-delay trans-delay-interval | dead dead-interval |  
[ simple [ plain plain-text | cipher cipher-text ] | { md5 | hmac-md5 } [ key-id  
{ plain plain-text | cipher cipher-text } ] | authentication-null | keychain  
keychain-name ] ] *
```

A virtual link is created.

This command must also be configured on the neighboring router.

---End

Follow-up Procedure

After virtual links are created, different default MTUs may be used on devices provided by different vendors. To ensure consistency, the MTU is set to 0 by default when the interface sends DD packets. For details, see [Configuring an Interface to Fill in the DD Packet with the Actual MTU](#).

5.3.4 (Optional) Configuring a Route Selection Rule on the router

You can configure the router to comply with the route selection rule defined in RFC 1583 or RFC 2328.

Context

RFC 2328 and RFC 1583 define the route selection rule differently. After OSPF is enabled on the router, specify a route selection rule based on the router configuration. The router complies with the route selection rule defined in RFC 1583 by default. If the neighboring router complies with the route selection rule defined in RFC 2328, configure the local router to comply with that defined in RFC 2328. This allows all routers in the OSPF area to comply with the same route selection rule.

Perform the following steps on the router running OSPF.

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
ospf [ process-id ]
```

The OSPF process view is displayed.

Step 3 Run:

```
undo rfc1583 compatible
```

The router is configured to comply with the route selection rule defined in RFC 2328, not RFC 1583.

By default, the router complies with route selection rule defined in RFC 1583.

----End

5.3.5 (Optional) Setting the OSPF Priority

When multiple routing protocols are used to select routes, you can set the OSPF priority to maneuver route selection.

Context

The routing protocols may share and select the routing information because the router may run multiple dynamic routing protocols at the same time. The system sets a priority for each routing protocol. When multiple routing protocols are used to select routes, the route selected by the routing protocol with a higher priority takes effect.

Perform the following steps on the router running OSPF.

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
ospf [ process-id ]
```

The OSPF process view is displayed.

Step 3 Run:

```
preference [ ase ] { preference | route-policy route-policy-name } *
```

The OSPF priority is set.

- **ase**: sets the priority of the AS-External route.
- *preference*: sets the priority for OSPF. The smaller the value, the higher the priority.
- *route-policy-name*: sets the priority for specified routes in the routing policy.

The default OSPF priority value is 10. When an ASE is specified, the default OSPF priority value is 150.

----End

5.3.6 (Optional) Restricting the Flooding of LSA Update Packets

When a large number of LSA update packets are flooded, the neighboring router may be busy processing LSA update packets and has to discard the Hello packets that are used to maintain neighbor relationships. This causes neighbor relationships to be interrupted. To resolve this problem, you can restrict the flooding of LSA update packets to maintain neighbor relationships.

Context

When multiple neighboring routers are configured or a large number of LSA update packets are flooded, the neighboring router may receive a large number of LSA update packets in a short period. This keeps the neighboring router busy processing a burst of LSA update packets and causes the neighboring router to unexpectedly discard Hello packets that are used to maintain the OSPF neighbor relationships. As a result, the neighbor relationships are interrupted. After the neighbor relationships are reestablished, more packets are to be exchanged. This intensifies neighbor relationship interruption. To resolve this problem, you can restrict the flooding of LSA update packets to maintain neighbor relationships.

Perform the following steps on the router running OSPF.

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
ospf [ process-id ]
```

The OSPF process view is displayed.

Step 3 Run:

```
flooding-control [ number transmit-number | timer-interval transmit-interval ] *
```

The flooding of LSA update packets is restricted.

By default, the number of LSA update packets to be flooded each time is 50, and the interval at which LSA update packets are flooded is 30s.

After the **flooding-control** command is run, the flooding of LSA update packets is immediately restricted.

If the **flooding-control** command is not run, the function of restricting the flooding of LSA update packets automatically takes effect when the number of neighboring routers exceeds 256.

----End

5.3.7 (Optional) Configuring the Maximum Number of Packet Retransmission Attempts

When no response to DD packets, LSU packets, or LSR packets is received, the retransmission mechanism is used and the maximum number of packet retransmission attempts is set.

Context

If no response is received when the maximum number of packet retransmission attempts is reached, the neighbor relationship will be interrupted.

By default, the retransmission mechanism is disabled.

Perform the following steps on the router running OSPF.

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
ospf [ process-id ]
```

The OSPF process view is displayed.

Step 3 Run:

```
retransmission-limit [ max-number ]
```

The maximum number of OSPF packet retransmission attempts is set.

max-number specifies the maximum number of packet retransmission attempts and is 30 by default.

---End

5.3.8 (Optional) Setting an Interval at Which an LSA Packet Is Retransmitted to the Neighboring router

You can control packet retransmission and improve the convergence rate by setting an interval at which an LSA packet is retransmitted to the neighboring router.

Context

After sending an LSA packet to the neighboring router, the router waits for a response. If no response is received within the set interval, the router retransmits the LSA packet to the neighboring router.

Perform the following steps on the router running OSPF.

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
interface interface-type interface-number
```

The interface view is displayed.

Step 3 Run:

```
ospf timer retransmit interval
```

An interval at which an LSA packet is retransmitted to the neighboring router is set.

Setting the interval to a proper value is recommended. A rather small interval will cause unnecessary retransmission. The interval is generally longer than a round trip of one packet transmitted between two routers.

The default retransmission interval is 5s and is widely used.

----End

5.3.9 (Optional) Configuring an Interface to Fill in a DD Packet with the Interface MTU

You can configure an interface to fill in the Interface MTU field of a DD packet with the interface MTU.

Context

The default MTU is 0.

After virtual links are created, different default MTUs may be used on devices provided by different vendors. To ensure consistency, the MTU is set to 0 by default when the interface sends DD packets.



CAUTION

Setting the MTU in a DD packet will have the neighbor relationship reestablished.

Perform the following steps on the router running OSPF.

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
interface interface-type interface-number
```

The interface view is displayed.

Step 3 Run:

```
ospf mtu-enable
```

The interface is configured to fill in a DD packet with the interface MTU and check whether the MTU in the DD packet from the neighboring router exceeds the MTU of the local router.

----End

5.3.10 Checking the Configuration

After basic OSPF functions are successfully configured, you can check information about the LSDB, neighbors in each area, and routing table.

Prerequisites

All configurations of basic OSPF functions are complete.

Procedure

- Run the **display ospf [process-id] peer** command to check OSPF neighbor information.
- Run the **display ospf [process-id] routing** command to check OSPF routing table information.
- Run the **display ospf [process-id] lsdb** command to check OSPF LSDB information.

----End

5.4 Configuring OSPF on the NBMA or P2MP Network

This section describes how to configure OSPF and modify attributes on the NBMA or point-to-multipoint (P2MP) network to flexibly construct the OSPF network.

5.4.1 Establishing the Configuration Task

To implement OSPF functions, configure OSPF on the NBMA or P2MP network.

Applicable Environment

As shown in [Table 5-1](#), OSPF classifies networks into four types based on the types of link layer protocols.



NOTE

Differentiated OSPF configurations that are applicable to the NBMA network and P2MP network are provided in this section. The OSPF configurations not provided here are applicable to the four types of networks.

Table 5-1 Network types supported by OSPF

Network Type	Characteristic	Default Configuration
Broadcast	On the broadcast network, Hello packets, LSU packets, and LSAck packets are multicasted; DD packets and LSR packets are unicasted.	If the link layer protocol is Ethernet or Fiber Distributed Data Interface (FDDI), OSPF regards the network as a broadcast network by default.

Network Type	Characteristic	Default Configuration
Non-broadcast multiple access (NBMA)	On an NBMA network, Hello packets, DD packets, LSR packets, LSU packets, and LSAck packets are unicasted. The NBMA network must be fully meshed. Any two routers on the NBMA network must be directly reachable.	If the link layer protocol is ATM, OSPF regards the network as an NBMA network by default.
Point-to-point (P2P)	On a P2P network, Hello packets, DD packets, LSR packets, LSU packets, and LSAck packets are multicasted.	If the link layer protocol is PPP, HDLC, or Link Access Procedure Balanced (LAPB), OSPF regards the network as a P2P network by default.
Point-to-multipoint (P2MP)	On a P2MP network, Hello packets are multicasted; DD packets, LSR packets, LSU packets, and LSAck packets are unicasted. The mask lengths of the routers on the P2MP network must be the same.	OSPF does not regard a network as a P2MP network by default regardless of any link layer protocol. A P2MP network is forcibly changed from the network of another type.

As shown in [Table 5-1](#), OSPF sends packets in different manners on networks of different types. Therefore, the difference between OSPF configurations on the networks lies in the packet sending configurations.

Pre-configuration Tasks

Before configuring OSPF on the NBMA or P2MP network, complete the following tasks:

- Configuring IP addresses for interfaces to ensure that neighboring routers are reachable at the network layer
- [Configuring Basic OSPF Functions](#)

Data Preparation

To configure OSPF on the NBMA or P2MP network, you need the following data.

No.	Data
1	Number of the interface running OSPF
2	Network type
3	DR priority of an interface
4	IP address of a neighbor on an NBMA network

No.	Data
5	Interval at which Hello packets are sent on an NBMA network

5.4.2 Configuring Network Types for OSPF Interfaces

OSPF classifies networks into four types based on the types of link layer protocols. You can configure the network type for an OSPF interface to forcibly change its original network type.

Context

By default, the physical interface type determines the network type.

- The network type of an Ethernet interface is Broadcast.
- The network type of a Frame Relay (FR) interface is NBMA.



NOTE

A P2MP network is forcibly changed from another other type of network.

The network types of the interfaces on both ends of a link must be the same; otherwise, the OSPF neighbor relationship cannot be established. Only when the network type of one OSPF interface is broadcast and the network type of the other OSPF interface is P2P, the two interfaces can still set up the neighbor relationship. but cannot learn the OSPF routing information each other.

Perform the following steps on the router running OSPF.

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
interface interface-type interface-number
```

The interface view is displayed.

Step 3 Run:

```
ospf network-type { broadcast | nbma | p2mp | p2p }
```

The network type of the OSPF interface is configured.

When the network type is configured for an interface, the original network type of the interface is replaced.

The network type can be configured based on the real world situations.

- On an interface with the broadcast network type, if a router that does not support the multicast address exists, change the network type of the interface to NBMA.
- On an interface with the NBMA network type, if the network is fully meshed or any two routers are directly connected, change the network type of the interface to broadcast and do not configure neighboring router information on the interface.
- On an interface with the NBMA network type, if the network is not fully meshed, change the network type of the interface to P2MP. After that, two indirectly connected routers can

communicate through one router that can directly reach both the two routers. After the network type of the interface is changed to P2MP, configuring neighboring router information on the interface is unnecessary.

- If only two routers run OSPF on the same network segment, changing the network type of the interface to P2P is recommended.



NOTE

OSPF cannot be configured on a null interface.

---End

5.4.3 Configuring NBMA Network Attributes

To implement OSPF functions, configure NBMA network attributes.

Procedure

Step 1 (Optional) Set the network type to NBMA.

The NBMA network must be fully meshed. Any two routers on the NBMA network must be directly reachable. In most cases, however, this requirement cannot be met. To resolve this problem, run specific commands to forcibly change the network type to NBMA. For details, see [Configuring Network Types for OSPF Interfaces](#).

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
interface interface-type interface-number
```

The interface view is displayed.

3. Run:

```
ospf network-type nbma
```

The network type of the OSPF interface is set to NBMA.

Step 2 (Optional) Set the interval at which Hello packets for polling are sent on the NBMA network.

On the NBMA network, after the neighbor relationship becomes invalid, the router sends Hello packets at an interval defined in the polling mechanism.

1. Run:

```
ospf timer poll interval
```

The interval at which Hello packets for polling are sent by an NBMA interface is set.

The default value is 120, in seconds.

Step 3 Configure a neighboring router on the NBMA network.

The interface with the network type of NBMA cannot broadcast Hello packets to discover neighboring routers. Therefore, the IP address of a neighboring router must be configured on the interface and whether the neighboring router can participate in DR election must be determined on the interface.

1. Run:

```
quit
```

- Exit from the interface view.
2. Run:
`ospf [process-id]`
The OSPF process view is displayed.
 3. Run:
`peer ip-address [dr-priority priority]`
A neighboring router is configured.
- End

5.4.4 Configuring P2MP Network Attributes

To implement OSPF functions, configure P2MP network attributes.

Context

Perform the following steps on the router running OSPF.

Procedure

Step 1 Disable OSPF from checking the network mask.

The OSPF neighbor relationship cannot be established between the routers with different mask lengths on the P2MP network. After OSPF is disabled from checking the network mask, the OSPF neighbor relationship can be properly established.

1. Run:
`system-view`
The system view is displayed.
2. Run:
`interface interface-type interface-number`
The interface view is displayed.
3. Run:
`ospf network-type p2mp`
The network type of the OSPF interface is configured.
A P2MP network is forcibly changed from another other type of network. For details, see [Configuring Network Types for OSPF Interfaces](#).
4. Run:
`ospf p2mp-mask-ignore`
OSPF is disabled from checking the network mask on the P2MP network.

Step 2 (Optional) Configure the router to filter the LSA packets to be sent.

When multiple links exist between two routers, you can configure the local router to filter the LSA packets to be sent. This can reduce unnecessary LSA retransmission attempts and save bandwidth resources.

1. Run:
`quit`

Exit from the interface view.

2. Run:
`ospf [process-id]`

The OSPF process view is displayed.

3. Run:
`filter-lsa-out peer ip-address { all | { summary [acl{ acl-number | acl-name }] | ase [acl{ acl-number | acl-name }] | nssa [acl{ acl-number | acl-name }] } * }`

The local router is configured to filter the LSA packets to be sent on the P2MP network.

By default, the LSA packets to be sent are not filtered.

----End

5.4.5 Checking the Configuration

After OSPF attributes on the NBMA network and P2MP network are set, you can check OSPF statistics, LSDB information, neighbor information, and interface information.

Prerequisites

The configurations for OSPF attributes on the NBMA network and P2MP network are complete.

Procedure

- Run the either of the following command to check LSDB information.
 - `display ospf [process-id] lsdb [brief]`
 - `display ospf [process-id] lsdb [router | network | summary | asbr | ase | nssa | opaque-link | opaque-area | opaque-as] [link-state-id] [originate-router [advertising-router-id] | self-originate] [age { min-value min-age-value | max-value max-age-value } *]`
- Run the `display ospf [process-id] peer [[interface-type interface-number] neighbor-id | brief | last-nbr-down]` command to view neighbor information.
- Run the `display ospf [process-id] nexthop` command to check next hop information.
- Run the either of the following command to check routing table information.
 - `display ospf [process-id] routing [ip-address [mask | mask-length]] [interface interface-type interface-number] [nexthop nexthop-address]`
 - `display ospf [process-id] routing router-id [router-id]`
- Run the `display ospf [process-id] interface [all | interface-type interface-number] [verbose]` command to check interface information.

----End

5.5 Configuring an OSPF Route Selection Rule

You can configure an OSPF route selection rule to meet requirements of complex networks.

5.5.1 Establishing the Configuration Task

Before configuring an OSPF route selection rule, familiarize yourself with the applicable environment, complete the pre-configuration tasks, and obtain the data required for the configuration. This will help you complete the configuration task quickly and efficiently.

Applicable Environment

In real world situations, you can configure an OSPF route selection rule by setting OSPF route attributes to meet the requirements of complex networks.

- Set the cost of an interface. The link connected to the interface with a smaller cost value preferentially transmits routing information.
- Configure equal-cost routes to implement load balancing.
- Configure a stub router during the maintenance operations such as upgrade to ensure stable data transmission through key routes.
- Suppress interfaces from sending or receiving packets to help select the optimal route.

Pre-configuration Tasks

Before configuring an OSPF route selection rule, complete the following tasks:

- Configuring IP addresses for interfaces to ensure that neighboring routers are reachable at the network layer
- [Configuring Basic OSPF Functions](#)

Data Preparation

To configure an OSPF route selection rule, you need the following data.

No.	Data
1	Interface cost
2	Maximum number of equal-cost routes
3	Equal-cost route preference

5.5.2 Setting the Interface Cost

You can adjust and optimize route selection by setting the OSPF interface cost.

Context

After the OSPF interface costs are set, the interface with a smaller cost value preferentially transmits routing information. This helps select the optimal route.

The OSPF interface cost can be set manually or calculated based on the interface bandwidth.

Perform the following steps on the router running OSPF.

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
interface interface-type interface-number
```

The interface view is displayed.

Step 3 Run:

```
ospf cost cost
```

The OSPF interface cost is set.

The router generally transmits routing information using the link connected to the interface with a smaller cost value.

If no interface cost is configured, the system automatically calculates the interface cost based on the interface bandwidth. The calculation formula is as follows: Cost of the interface = Bandwidth reference value/Interface bandwidth. The integer of the calculated result is the cost of the interface. If the calculated result is smaller than 1, the cost value is 1. By default, the bandwidth reference value is 100, in Mbit/s. Changing the bandwidth reference value can change the cost of an interface.

Perform the following steps to change the bandwidth reference value:

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
ospf [ process-id ]
```

The OSPF process view is displayed.

3. Run:

```
bandwidth-reference value
```

The bandwidth reference value is set.

Ensure that the bandwidth reference values of routers in an OSPF process are the same.

----End

5.5.3 Configuring Equal-Cost Routes

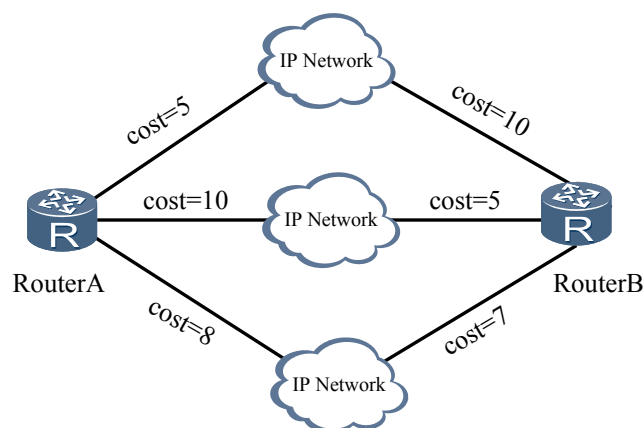
You can set the number of OSPF equal-cost routes and route preference to implement load balancing and adjust route selection.

Context

If the destinations and costs of the multiple routes discovered by one routing protocol are the same, load balancing can be implemented among the routes.

As shown in [Figure 5-4](#), three routes between router A and router B that run OSPF have the same costs. The three routes are equal-cost routes for load balancing.

Figure 5-4 Networking diagram of equal-cost routes



Perform the following steps on the router running OSPF.

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
ospf [ process-id ]
```

The OSPF process view is displayed.

Step 3 Run:

```
maximum load-balancing number
```

The maximum number of equal-cost routes is set.

NOTE

The maximum number of equal-cost routes is 4, by default, it is 4.

Step 4 (Optional) Run:

```
nexthop ip-address weight value
```

The route preferences are configured for load balancing.

When the number of equal-cost routes on the live network is greater than that specified in the **maximum load-balancing** command, valid routes are randomly selected for load balancing. To specify valid routes for load balancing, run the **nexthop** command to set the route preference. Ensure that the preferences of valid routes to be used must be high.

The smaller the **weight** value, the higher the preference of the route. The default **weight** value is 255, which indicates that load balancing is implemented regardless of the route preferences.

---End

5.5.4 Configuring a Stub Router

To ensure that a route is not interrupted during flapping-triggering maintenance operations such as upgrade, you can configure a router as a stub router to allow traffic to bypass the route on the stub router.

Context

After a stub router is configured, the route on the stub router will not be preferentially selected. After the route cost is set to the maximum value 65535, traffic generally bypasses the router. This ensures an uninterrupted route on the router during maintenance operations such as upgrade.

Perform the following steps on the router running OSPF.

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
ospf [ process-id ]
```

The OSPF process view is displayed.

Step 3 Run:

```
stub-router [ on-startup [ interval ] ]
```

A stub router is configured.

By default, no router is configured as a stub router.

If a router is configured as a stub router, the router keeps functioning as a stub router for 500s.

 **NOTE**

The stub router configured in this manner is irrelevant to the router in the stub area.

---End

5.5.5 Suppressing an Interface from Receiving and Sending OSPF Packets

After an interface is suppressed from receiving and sending OSPF packets, routing information can bypass a specific router and the local router can reject routing information advertised by another router.

Context

Suppressing an interface from receiving and sending OSPF packets helps routing information to bypass a specific router and enables the local router to reject routing information advertised by another router. This ensures that an optimal route is provided.

Perform the following steps on the router running OSPF.

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
ospf [ process-id ]
```

The OSPF process view is displayed.

Step 3 Run:

```
silent-interface { all | interface-type interface-number }
```

An interface is suppressed from receiving and sending OSPF packets.

The same interface in different processes can be suppressed from sending and receiving OSPF packets, but the **silent-interface** command is valid only for the OSPF interface in the local process.

After an OSPF interface is configured to be in the silent state, the interface can still advertise its direct routes. Hello packets on the interface, however, cannot be forwarded. Therefore, no neighbor relationship can be established on the interface. This can enhance the networking adaptability of OSPF and reduce system resource consumption.

---End

5.5.6 Checking the Configuration

After an OSPF route selection rule is configured, you can check information about the OSPF routing table, interface, and next hop.

Prerequisites

All OSPF route selection configurations are complete.

Procedure

- Run the **display ospf [process-id] routing [ip-address [mask | mask-length]] [interface interface-type interface-number] [nexthop nexthop-address]** command to check the OSPF routing table information.
- Run the **display ospf [process-id] interface [all | interface-type interface-number] [verbose]** command to check OSPF interface information.

---End

5.6 Controlling OSPF Routing Information

You can control the advertising and receiving of OSPF routing information and import routes of other protocols.

5.6.1 Establishing the Configuration Task

Before controlling OSPF routing information, familiarize yourself with the applicable environment, complete the pre-configuration tasks, and obtain the data required for the configuration. This will help you complete the configuration task quickly and efficiently.

Applicable Environment

You can control the advertising and receiving of OSPF routing information and import routes of other protocols.

Pre-configuration Tasks

Before controlling OSPF routing information, complete the following tasks:

- Configuring IP addresses for interfaces to ensure that neighboring routers are reachable at the network layer
- [Configuring Basic OSPF Functions](#)

Data Preparation

To control OSPF routing information, you need the following data.

No.	Data
1	Link cost
2	ACL for route filtering
3	Name of the imported routing protocol, OSPF process ID, and default parameters

5.6.2 Configuring OSPF to Import External Routes

Importing the routes discovered by other routing protocols can enrich OSPF routing information.

Context

To access a router running a non-OSPF protocol, an OSPF-capable router needs to import routes of the non-OSPF protocol into the OSPF network.

OSPF provides loop-free intra-area routes and inter-area routes; however, OSPF cannot prevent external routing loops. Therefore, exercise caution when configuring OSPF to import external routes. For details, see "OSPF VPN Extension" in the *Huawei AR150&200 Series Enterprise Routers Feature Description - VPN*.

Perform the following steps on the ASBR running OSPF.

Procedure

- Step 1** Run:
- ```
system-view
```

The system view is displayed.

**Step 2** Run:

```
ospf [process-id]
```

The OSPF process view is displayed.

**Step 3** Run:

```
import-route { limit limit-number | { bgp [permit-ibgp] | direct | unr | rip
[process-id-rip] | static | isis [process-id-isis] | ospf [process-id-ospf] }
[cost cost | type type | tag tag | route-policy route-policy-name] * }
```

Routes are imported from another protocol.

**Step 4** (Optional) Run:

```
default { cost { cost | inherit-metric } | limit limit | tag tag | type type } *
```

The default values of parameters (the cost, number of routes, tag, and type) are set for imported routes.

When OSPF imports external routes, you can set default values for some additional parameters, such as the cost, number of routes to be imported, route tag, and route type. The route tag is used to identify the protocol-related information. For example, it can be used to differentiate AS numbers carried in BGP routes imported by OSPF.

By default, the cost of the external routes imported by OSPF is 1; a maximum of 2147483647 routes can be imported each time; the type of the imported external routes is Type 2; the default tag value of the imported routes is 1.

 **NOTE**

You can run one of the following commands to set the cost of the imported route. The following commands are listed in descending order of priorities.

- Run the **apply cost** command to set the cost of a route.
- Run the **import-route** command to set the cost of the imported route.
- Run the **default** command to set the default cost of the imported route.

**Step 5** (Optional) Run:

```
filter-policy { acl-number | acl-nameacl-name | ip-prefixip-prefix-name } export
[protocol [process-id]]
```

Routes imported using [Step 3](#) can be advertised only when meeting filtering conditions.

OSPF filters the imported routes. OSPF uses Type 5 LSAs to carry routes that meet the filtering conditions and advertises these Type 5 LSAs.

You can specify the parameter *protocol [ process-id ]* to filter the routes of a certain routing protocol or a certain OSPF process. If *protocol [ process-id ]* is not specified, OSPF filters all imported routes.

The **import-route** command cannot be used to import the default route from another AS.

---End

## 5.6.3 Configuring OSPF to Import a Default Route

The default route is widely applied on the OSPF network to reduce routing entries in the routing table and filter specific routing information.

## Context

On the area border and AS border of an OSPF network generally reside multiple routers for next-hop backup or traffic load balancing. A default route can be configured to reduce routing entries and improve resource usage on the OSPF network.

The default route is generally applied to the following scenarios:

1. An ABR in an area advertises Type 3 LSAs carrying the default route within the area. routers in the area use the received default route to forward inter-area packets.
2. An ASBR in an AS advertises Type 5 or Type 7 LSAs carrying the default route within the AS. routers in the AS use the received default route to forward AS external packets.

When no exactly matched route is discovered, the router can forward packets through the default route.

The preference of the default route in Type 3 LSAs is higher than that of the route in Type 5 or Type 7 LSAs.

The advertising mode of the default route is determined by the type of the area to which the default route is imported, as shown in [Table 5-2](#).

**Table 5-2** Default route advertising mode

| Area Type    | Generated By                                       | Advertised By | LSA Type   | Flooding Area |
|--------------|----------------------------------------------------|---------------|------------|---------------|
| Common area  | The <b>default-route-advertise</b> command         | ASBR          | Type 5 LSA | Common area   |
| Stub area    | Automatically                                      | ABR           | Type 3 LSA | Stub area     |
| NSSA         | The <b>nssa[ default-route-advertise ]</b> command | ASBR          | Type 7 LSA | NSSA          |
| Totally NSSA | Automatically                                      | ABR           | Type 3 LSA | NSSA          |

Perform the following steps on the ASBR running OSPF.

## Procedure

**Step 1** Run:

```
system-view
```

The system view is displayed.

**Step 2** Run:

```
ospf [process-id]
```

The OSPF process view is displayed.

**Step 3** Run:

```
default-route-advertise [[always | permit-calculate-other] | cost cost | type
type | route-policy route-policy-name] *
```

The default route is imported into the OSPF process.

To configure the parameter *cost* to specify the default cost of Type-3 summary LSAs, enable VPN first.

Before advertising a default route, OSPF compares the preferences of default routes. Therefore, if a static default route is configured on an OSPF device, to add the default route advertised by OSPF to the current routing table, ensure that the preference of the configured static default route is lower than that of the default route advertised by OSPF.

For details about how to configure the default route in the NSSA, see [Configuring an NSSA](#).

---End

## 5.6.4 Configuring Route Summarization

When a large-scale OSPF network is deployed, you can configure route summarization to reduce routing entries. Otherwise, a large number of routing entries are generated and consume system resources unexpectedly.

### Context

Route summarization on a large-scale OSPF network efficiently reduces routing entries. This minimizes system resource consumption and maintains the system performance. In addition, if a specific link frequently alternates between Up and Down, the links not involved in the route summarization will not be affected. This prevents route flapping and improves the network stability.

Perform the following steps on the router running OSPF.

### Procedure

- Configure ABR route summarization.
  1. Run:

```
system-view
```

The system view is displayed.
  2. Run:

```
ospf [process-id]
```

The OSPF process view is displayed.
  3. Run:

```
area area-id
```

The OSPF area view is displayed.
  4. Run:

```
abr-summary ip-address mask [[advertise | not-advertise] | cost cost]
*
```

ABR route summarization is configured.
- Configure ASBR route summarization.
  1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
ospf [process-id]
```

The OSPF process view is displayed.

3. Run:

```
asbr-summary ip-address mask [not-advertise | tag tag | cost cost |
distribute-delay interval] *
```

ASBR route summarization is configured.



**NOTE**

After route summarization is configured, the routing table on the local OSPF router remains the same. The routing table on another OSPF router, however, contains only one summarized route, no specific route. This summarized route is not removed until all specific routes are interrupted.

---End

## 5.6.5 Configuring OSPF to Filter Routes Received by OSPF

By configuring filtering conditions for the received routes, you can allow only the routes that meet the filtering conditions to be added to the routing table.

### Context

Perform the following steps on the router running OSPF.

### Procedure

**Step 1** Run:

```
system-view
```

The system view is displayed.

**Step 2** Run:

```
ospf [process-id]
```

The OSPF process view is displayed.

**Step 3** Run:

```
filter-policy { acl-number | acl-name acl-name | ip-prefix ip-prefix-name } import
```

Routes received by OSPF are filtered.

OSPF is a dynamic routing protocol based on the link state, and routing information is carried in LSAs. The **filter-policy import** command cannot be used to filter the advertised and received LSAs. Actually, this command is used to filter the routes calculated by OSPF. Only the routes that meet the filtering conditions are added to the routing table. Therefore, the LSDB is not affected regardless of whether the received routes meet the filtering conditions.

---End

## 5.6.6 Configuring the router to Filter LSAs to Be Sent

Filtering the LSAs to be sent on the local router can prevent unnecessary LSA transmission. This reduces the size of the LSDB on the neighboring router and speeds up network convergence.

## Context

When multiple links exist between two routers, you can configure the local router to filter the LSAs to be sent. This prevents unnecessary LSA transmission and saves bandwidth resources.

Perform the following steps on the router running OSPF.

## Procedure

### Step 1 Run:

```
system-view
```

The system view is displayed.

### Step 2 Run:

```
interface interface-type interface-number
```

The interface view is displayed.

### Step 3 Run:

```
ospf filter-lsa-out { all | { summary [acl { acl-number | acl-name }] | ase
[acl { acl-number | acl-name }] | nssa [acl { acl-number | acl-name }] } * }
```

The LSAs to be sent are filtered.

By default, the LSAs to be sent are not filtered.

---End

## 5.6.7 (Optional) Configuring OSPF to Filter LSAs in an Area

Filtering LSAs in an area can prevent unnecessary LSA transmission. This reduces the size of the LSDB on the neighboring router and speeds up network convergence.

## Context

After filtering conditions are set for the incoming or outgoing Type 3 LSAs (Summary LSAs) in an area, only the Type 3 LSAs that meet the filtering conditions can be received or advertised.

This function is applicable only to the ABR.

Perform the following steps on the router running OSPF.

## Procedure

### Step 1 Run:

```
system-view
```

The system view is displayed.

### Step 2 Run:

```
ospf [process-id]
```

The OSPF process view is displayed.

### Step 3 Run:

```
area area-id
```



The OSPF area view is displayed.

**Step 4** Filter incoming or outgoing Type 3 LSAs in the area.

- Filter incoming Type 3 LSAs in the area.

Run the **filter** { *acl-number* | **acl-name** *acl-name* | **ip-prefix** *ip-prefix-name* | **route-policy** *route-policy-name* } **export** command to filter incoming Type 3 LSAs in the area.

- Filter outgoing Type 3 LSAs in the area.

Run the **filter** { *acl-number* | **acl-name** *acl-name* | **ip-prefix** *ip-prefix-name* | **route-policy** *route-policy-name* } **import** command to filter outgoing Type 3 LSAs in the area.

----End

## 5.6.8 (Optional) Enabling the Mesh-Group Function

The mesh-group function is used to prevent repeated flooding and save system resources.

### Context

When concurrent links exist between two routers, you can enable the mesh-group function to reduce the load on the links.

The neighboring router ID identifies each mesh group. Several concurrent links are added to a mesh group. Flooding is implemented once in the group. You can add interfaces that meet the following conditions to the same mesh group.

- The interfaces belong to the same area and OSPF process.
- The interfaces begin to exchange DD packets.
- The interfaces are connected to the same neighboring router.

Perform the following steps on the router running OSPF.

### Procedure

**Step 1** Run:

```
system-view
```

The system view is displayed.

**Step 2** Run:

```
ospf [process-id]
```

The OSPF process view is displayed.

**Step 3** Run:

```
mesh-group enable
```

The mesh-group function is enabled.

By default, the mesh-group function is disabled.

----End

## 5.6.9 Setting the Maximum Number of External LSAs in the LSDB

You can set the maximum number of external LSAs in the LSDB to keep a proper number of external LSAs.

## Procedure

- Step 1** Run:  
`system-view`  
The system view is displayed.
- Step 2** Run:  
`ospf [ process-id ]`  
The OSPF process view is displayed.
- Step 3** Run:  
`lsdb-overflow-limit number`  
The maximum number of external LSAs in the LSDB is set.
- End

### 5.6.10 Checking the Configuration

After controlling OSPF routing information, you can check information about the OSPF routing table, interface, and ASBR summarization.

#### Prerequisites

The configurations of controlling OSPF routing information are complete.

#### Procedure

- Run either of the following commands to check routing table information.
    - `display ospf [ process-id ] routing [ ip-address [ mask | mask-length ] ] [ interface interface-type interface-number ] [ nexthop nexthop-address ]`
    - `display ospf [ process-id ] routing router-id [ router-id ]`
  - Run the `display ospf [ process-id ] interface [ all | interface-type interface-number ] [ verbose ]` command to check OSPF interface information.
  - Run the `display ospf [ process-id ] asbr-summary [ ip-address mask ]` command to check OSPF ASBR summarization information.
- End

## 5.7 Configuring an OSPF Stub Area

Configuring a non-backbone area as a stub area can reduce routing entries in the area in an AS does not transmit routes learned from other areas in the AS or AS external routes. This reduces bandwidth and storage resource consumption.

#### Applicable Environment

The number of LSAs can be reduced by partitioning an AS into different areas. To reduce the number of entries in the routing table and the number of LSAs to be transmitted in a non-backbone area, configure the non-backbone area on the border of the AS as a stub area.

Configuring a stub area is optional. A stub area generally resides on the border of an AS. For example, a non-backbone area with only one ABR can be configured as a stub area. In a stub

area, the number of entries in the routing table and the amount of routing information to be transmitted greatly decrease.

Note the following points when configuring a stub area:

- The backbone area (Area 0) cannot be configured as a stub area.
- If an area needs to be configured as a stub area, all the routers in this area must be configured with stub attributes using the **stub** command.
- An ASBR cannot exist in a stub area. External routes are not transmitted in the stub area.
- Virtual links cannot exist in the stub area.

## Pre-configuration Tasks

Before configuring a stub area, complete the following tasks:

- Configuring IP addresses for interfaces to ensure that neighboring routers are reachable at the network layer
- [Configuring Basic OSPF Functions](#)

## Data Preparation

To configure a stub area, you need the following data.

| No. | Data                                                                                                                                     |
|-----|------------------------------------------------------------------------------------------------------------------------------------------|
| 1   | (Optional) Cost of the default route to the stub area<br><b>NOTE</b><br>By default, the cost of the default route to the stub area is 1. |

## Procedure

**Step 1** Run:

```
system-view
```

The system view is displayed.

**Step 2** Run:

```
ospf [process-id]
```

The OSPF process view is displayed.

**Step 3** Run:

```
area area-id
```

The OSPF area view is displayed.

**Step 4** Run:

```
stub
```

The specified area is configured as a stub area.

 **NOTE**

- All routers in a stub area must be configured with stub attributes using the **stub** command.
- Configuring or deleting stub attributes will update routing information in the area. Stub attributes can be deleted or configured again only after the routing update is complete.

**Step 5** (Optional) Run:

```
stub [no-summary]
```

The ABR is prevented from sending Type 3 LSAs to the stub area.

**Step 6** (Optional) Run:

```
default-cost cost
```

The cost of the default route to the stub area is set.

To ensure the reachability of AS external routes, the ABR in the stub area generates a default route and advertises the route to the non-ABR routers in the stub area.

By default, the cost of the default route to the stub area is 1.

----End

## Checking the Configuration

Run either of the following commands to check LSDB information.

- **display ospf** [ *process-id* ] **lsdb** [ **brief** ]
- **display ospf** [ *process-id* ] **lsdb** [ **router** | **network** | **summary** | **asbr** | **ase** | **nssa** | **opaque-link** | **opaque-area** | **opaque-as** ] [ *link-state-id* ] [ **originate-router** [ *advertising-router-id* ] | **self-originate** ] [ **age** { **min-value** *min-age-value* | **max-value** *max-age-value* } \* ]

Run either of the following commands to check routing table information.

- **display ospf** [ *process-id* ] **routing** [ *ip-address* [ *mask* | *mask-length* ] ] [ **interface** *interface-type interface-number* ] [ **nexthop** *nexthop-address* ]
- **display ospf** [ *process-id* ] **routing router-id** [ *router-id* ]

Run the **display ospf** [ *process-id* ] **abr-asbr** [ *router-id* ] command to check ASBR and ABR information.

## 5.8 Configuring an NSSA

Configuring a non-backbone area on the border of an AS as an NSSA does not transmit routes learned from other areas in the AS but imports AS external routes. This reduces bandwidth and storage resource consumption on the router.

### Applicable Environment

An NSSA is configured in the scenario where AS external routes are to be imported but not forwarded to save system resources.

The NSSA is a new type of OSPF area. Neither the NSSA nor the stub area transmits routes learned from other areas in the AS it resides. The stub area does not allow AS external routes to be imported, whereas the NSSA allows AS external routes to be imported and forwarded in the entire AS.

Type 7 LSAs are used to carry imported AS external routing information in the NSSA. Type 7 LSAs are generated by the ASBRs of NSSAs and flooded only in the NSSAs where ASBRs reside. The ABR in an NSSA selects certain Type 7 LSAs from the received ones and translates them into Type 5 LSAs to advertise AS external routing information to the other areas over the OSPF network.

To configure an area as an NSSA, configure NSSA attributes on all the routers in this area.

## Pre-configuration Tasks

Before configuring an NSSA, complete the following tasks:

- Configuring IP addresses for interfaces to ensure that neighboring routers are reachable at the network layer
- [Configuring Basic OSPF Functions](#)

## Data Preparation

To configure an NSSA, you need the following data.

| No. | Data                                                                                                                           |
|-----|--------------------------------------------------------------------------------------------------------------------------------|
| 1   | (Optional) Cost of the default route to the NSSA<br><b>NOTE</b><br>By default, the cost of the default route to the NSSA is 1. |

## Procedure

### Step 1 Run:

```
system-view
```

The system view is displayed.

### Step 2 Run:

```
ospf [process-id]
```

The OSPF process view is displayed.

### Step 3 Run:

```
area area-id
```

The OSPF area view is displayed.

### Step 4 Run:

```
nssa [default-route-advertise | flush-waiting-timer interval-value | no-import-route | no-summary | set-n-bit | suppress-forwarding-address | translator-always | translator-interval interval-value | zero-address-forwarding] *
```

The specified area is configured as an NSSA.

#### NOTE

- All routers in the NSSA must be configured with NSSA attributes using the **nssa** command.
- Configuring or deleting NSSA attributes may trigger routing update in the area. A second configuration of NSSA attributes can be implemented or canceled only after routing update is complete.

The **nssa** command is applicable to the following scenarios:

- The parameter **default-route-advertise** is used to advertise Type 7 LSAs carrying the default route on the ABR or ASBR to the NSSA.

Type 7 LSAs carrying the default route will be generated regardless of whether the default route 0.0.0.0 exists in the routing table on the ABR. On the ASBR, however, the default Type 7 LSA is generated only when the default route 0.0.0.0 exists in the routing table.

- When the area to which the ASBR belongs is configured as an NSSA, invalid Type 5 LSAs from other routers in the area where LSAs are flooded will be reserved. These LSAs will be deleted only when the aging time reaches 3600s. The router performance is affected because the forwarding of a large number of LSAs consumes the memory resources. To resolve such a problem, you can set the parameter **flush-waiting-timer** to the maximum value 3600s for Type 5 LSAs so that the invalid Type 5 LSAs from other routers can be deleted in time.

 **NOTE**

- When the LS age field value (aging time) in the header of an LSA reaches 3600s, the LSA is deleted.
- If an ASBR also functions as an ABR, **flush-waiting-timer** does not take effect. This prevents Type 5 LSAs in the non-NSSAs from being deleted.
- If an ASBR also functions as an ABR, set the parameter **no-import-route** to prevent external routes imported using the **import-route** command from being advertised to the NSSA.
- To reduce the number of LSAs that are transmitted to the NSSA, set the parameter **no-summary** on an ABR. This prevents the ABR from transmitting Type 3 LSAs to the NSSA.
- After the parameter **set-n-bit** is configured, the router re-establishes neighbor relationships with the neighboring routers in the NSSA.
- If multiple ABRs are deployed in the NSSA, the system automatically selects an ABR (generally the router with the largest router ID) as a translator to translate Type 7 LSAs into Type 5 LSAs. You can also configure the parameter **translator-always** on an ABR to specify the ABR as an all-the-time translator. To specify two ABRs for load balancing, configure the parameter **translator-always** on two ABRs to specify the ABRs as all-the-time translators. This prevents LSA flooding caused by translator role changes.
- The parameter **translator-interval** is used to ensure uninterrupted services when translator roles change. The *interval-value* value must be greater than the flooding period.

**Step 5** (Optional)Run:

```
default-cost cost
```

The cost of the default route to the NSSA is set.

To ensure the reachability of AS external routes, the ABR in the NSSA generates a default route and advertises the route to the other routers in the NSSA.

Type 7 LSAs can be used to carry default route information to guide traffic to other ASs.

Multiple ABRs may be deployed in an NSSA. To prevent routing loops, ABRs do not calculate the default routes advertised by each other.

By default, the cost of the default route to the NSSA is 1.

----End

## Checking the Configuration

Run either of the following commands to check LSDB information.

- **display ospf** [ *process-id* ] **lsdb** [ **brief** ]
- **display ospf** [ *process-id* ] **lsdb** [ **router** | **network** | **summary** | **asbr** | **ase** | **nssa** | **opaque-link** | **opaque-area** | **opaque-as** ] [ *link-state-id* ] [ **originate-router** [ *advertising-router-id* ] ] | **self-originate** ]

Run either of the following commands to check routing table information.

- **display ospf** [ *process-id* ] **routing** [ *ip-address* [ *mask* | *mask-length* ] ] [ **interface** *interface-type interface-number* ] [ **nexthop** *nexthop-address* ]
- **display ospf** [ *process-id* ] **routing router-id** [ *router-id* ]

Run the **display ospf** [ *process-id* ] **interface** [ **all** | *interface-type interface-number* ] [ **verbose** ] command to check OSPF interface information.

## 5.9 Configuring BFD for OSPF

After BFD for OSPF is enabled, when a link fails, the router rapidly detects the failure, notifies the OSPF process or interface of the fault, and instructs OSPF to recalculate routes. This speeds up OSPF network convergence.

### 5.9.1 Establishing the Configuration Task

Before configuring BFD for OSPF, familiarize yourself with the applicable environment, complete the pre-configuration tasks, and obtain the data required for the configuration. This will help you complete the configuration task quickly and efficiently.

#### Applicable Environment

OSPF enables the router to periodically send Hello packets to a neighboring router for fault detection. Detecting a fault takes more than 1s. As technologies develop, voice, video, and other VOD services are widely used. These services are quite sensitive to packet loss and delays. When traffic is transmitted at gigabit rates, long-time fault detection will cause packet loss. This cannot meet high reliability requirements of the carrier-class network.

BFD for OSPF is introduced to resolve this problem. After BFD for OSPF is configured in a specified process or on a specified interface, the link status can be rapidly detected and fault detection can be completed in milliseconds. This speeds up OSPF convergence when the link status changes.

#### NOTE

A BFD session currently does not detect route switching. If the change of bound peer IP address causes a route to switch to another link, the BFD session is negotiated again only when the original link fails.

#### Pre-configuration Tasks

Before configuring BFD for OSPF, complete the following task:

- Configuring IP addresses for interfaces to ensure that neighboring routers are reachable at the network layer
- [Configuring Basic OSPF Functions](#)

## Data Preparation

To configure BFD for OSPF, you need the following data.

| No. | Data                                                                                                        |
|-----|-------------------------------------------------------------------------------------------------------------|
| 1   | Number of the OSPF process to be enabled with BFD for OSPF                                                  |
| 2   | Type and number of the interface to be enabled with BFD for OSPF                                            |
| 3   | (Optional) Values of BFD session parameters<br><b>NOTE</b><br>The default parameter values are recommended. |

### 5.9.2 Configuring BFD for OSPF in a Specified Process

Configuring BFD for OSPF in a specified process helps the system to rapidly detect the link status and speeds up OSPF convergence in the case of a link failure.

#### Context

After BFD for OSPF is configured, when detecting a link fault, BFD rapidly notifies the routers on both ends of the link of the fault, triggering rapid OSPF convergence. When the OSPF neighbor relationship goes Down, the BFD session will be dynamically deleted.

Before configuring BFD for OSPF, enable BFD globally.

Perform the following steps on the routers between which a BFD session is to be created.

#### Procedure

**Step 1** Run:

```
system-view
```

The system view is displayed.

**Step 2** Run:

```
bfd
```

BFD is globally configured.

**Step 3** Run:

```
quit
```

Return to the system view.

**Step 4** Run:

```
ospf [process-id]
```

The OSPF process view is displayed.

**Step 5** Run:

```
bfd all-interfaces enable
```

BFD for OSPF is configured. The default parameter values are used to create a BFD session.



If all the interfaces in a certain process are configured with BFD and their neighbor relationships are in the Full state, OSPF creates BFD sessions with default parameter values on all the interfaces in the process.

### Step 6 (Optional) Run:

```
bfd all-interfaces { min-rx-interval receive-interval | min-tx-interval transmit-interval | detect-multiplier multiplier-value } *
```

BFD session parameters are modified.

You can skip this step. The default interval at which BFD packets are transmitted and the default detection multiplier are recommended.

The parameters are configured based on the network status and network reliability requirements. A short interval at which BFD packets are transmitted can be configured for a link that has a higher requirement for reliability. A long interval at which BFD packets are transmitted can be configured for a link that has a lower requirement for reliability.

#### NOTE

- Actual interval at which BFD packets are transmitted on the local router = Max { configured interval *transmit-interval* at which BFD packets are transmitted on the local router, configured interval *receive-interval* at which BFD packets are received on the peer router }
- Actual interval at which BFD packets are received on the local router = Max { configured interval *transmit-interval* at which BFD packets are transmitted on the peer router, configured interval *receive-interval* at which BFD packets are received on the local router }
- Actual time for detecting BFD packets = Actual interval at which BFD packets are received on the local router x Configured detection multiplier *multiplier-value* on the peer router

For example:

- On the local router, the configured interval at which BFD packets are transmitted is 200 ms; the configured interval at which BFD packets are received is 300 ms; the detection multiplier is 4.
- On the peer router, the configured interval at which BFD packets are transmitted is 100 ms; the interval at which BFD packets are received is 600 ms; the detection multiplier is 5.

Then:

- On the local router, the actual interval at which BFD packets are transmitted is 600 ms calculated by using the formula max {200 ms, 600 ms}; the interval at which BFD packets are received is 300 ms calculated by using the formula max {100 ms, 300 ms}; the detection period is 1500 ms calculated by multiplying 300 ms by 5.
- On the peer router, the actual interval at which BFD packets are transmitted is 300 ms calculated by using the formula max {100 ms, 300 ms}, the actual interval at which BFD packets are received is 600 ms calculated by using the formula max {200 ms, 600 ms}, and the detection period is 2400 ms calculated by multiplying 600 ms by 4.

### Step 7 (Optional) Prevent an interface from dynamically creating a BFD session.

After BFD for OSPF is configured, all interfaces on which neighbor relationships are Full in the OSPF process will create BFD sessions. To prevent specific interfaces from being enabled with BFD, disable these interfaces from dynamically creating BFD sessions.

1. Run:

```
quit
```

Return to the system view.

2. Run:

```
interface interface-type interface-number
```

The interface view is displayed.

3. Run:

```
ospf bfd block
```

An interface is prevented from dynamically creating a BFD session.

----End

## 5.9.3 Configuring BFD for OSPF on a Specified Interface

Configuring BFD for OSPF on a specified interface helps speed up OSPF convergence in the case of an interface failure.

### Context

After BFD for OSPF is configured on a specified interface and the interface becomes faulty, the router rapidly detects the fault and instructs OSPF to recalculate routes. This speeds up OSPF convergence. When the OSPF neighbor relationship goes Down, the BFD session between OSPF neighbors is dynamically deleted.

Before configuring BFD for OSPF, enable BFD globally.

Perform the following steps on the router:

### Procedure

**Step 1** Run:

```
system-view
```

The system view is displayed.

**Step 2** Run:

```
bfd
```

BFD is globally configured.

**Step 3** Run:

```
quit
```

Return to the system view.

**Step 4** Run:

```
interface interface-type interface-number
```

The interface view is displayed.

**Step 5** Run:

```
ospf bfd enable
```

BFD for OSPF is configured. The default parameter values are used to create a BFD session.

If all the interfaces in a certain process are configured with BFD and their neighbor relationships are in the Full state, OSPF creates BFD sessions with default parameter values on specified interfaces in the process.

#### NOTE

The priority of BFD for OSPF configured on an interface is higher than that of BFD for OSPF configured for a process.

**Step 6** (Optional) Run:

```
ospf bfd { min-rx-interval receive-interval | min-tx-interval transmit-interval |
detect-multiplier multiplier-value } *
```

BFD session parameters are modified.

You can skip this step. The default interval at which BFD packets are transmitted and the default detection multiplier are recommended.

The parameters are configured based on the network status and network reliability requirements. A short interval at which BFD packets are transmitted can be configured for a link that has a higher requirement for reliability. A long interval at which BFD packets are transmitted can be configured for a link that has a lower requirement for reliability.

#### NOTE

- Actual interval at which BFD packets are transmitted on the local router = Max { configured interval *transmit-interval* at which BFD packets are transmitted on the local router, configured interval *receive-interval* at which BFD packets are received on the peer router }
- Actual interval at which BFD packets are received on the local router = Max { configured interval *transmit-interval* at which BFD packets are transmitted on the peer router, configured interval *receive-interval* at which BFD packets are received on the local router }
- Actual time for detecting BFD packets = Actual interval at which BFD packets are received on the local router x Configured detection multiplier *multiplier-value* on the peer router

For example:

- On the local router, the configured interval at which BFD packets are transmitted is 200 ms; the interval at which BFD packets are received is set to 300 ms; the detection multiplier is 4.
- On the peer router, the configured interval at which BFD packets are transmitted is 100 ms; the interval at which BFD packets are received is 600 ms; the detection multiplier is 5.

Then:

- On the local router, the actual interval at which BFD packets are transmitted is 600 ms calculated by using the formula max {200 ms, 600 ms}; the interval at which BFD packets are received is 300 ms calculated by using the formula max {100 ms, 300 ms}; the detection period is 1500 ms calculated by multiplying 300 ms by 5.
- On the peer router, the actual interval at which BFD packets are transmitted is 300 ms calculated by using the formula max {100 ms, 300 ms}, the actual interval at which BFD packets are received is 600 ms calculated by using the formula max {200 ms, 600 ms}, and the detection period is 2400 ms calculated by multiplying 600 ms by 4.

---End

## 5.9.4 Checking the Configuration

After configuring BFD for OSPF, you can view information about the BFD session between two OSPF neighbors.

### Prerequisites

All BFD for OSPF configurations are complete.

### Procedure

- Run the **display ospf** [*process-id*] **bfd session** *interface-type interface-number* [*router-id*] or **display ospf** [*process-id*] **bfd session** { *router-id* | **all** } command to check information about the BFD session between two OSPF neighbors.

---End

## 5.10 Configuring OSPF GR

Configuring OSPF GR to avoid traffic interruption and route flapping caused by the active/standby switchover.

### 5.10.1 Establishing the Configuration Task

Before configuring OSPF GR, familiarize yourself with the applicable environment, complete the pre-configuration tasks, and obtain the data required for the configuration. This will help you complete the configuration task quickly and efficiently.

#### Applicable Environment

To avoid traffic interruption and route flapping caused by the active/standby switchover, you can enable OSPF GR.

After the OSPF process is restarted through GR, the Restarter and the Helper reestablish the neighbor relationship, exchange routing information, synchronize the LSDB, and update the routing table and forwarding table. These operations ensure the fast convergence of OSPF and the stability the network topology.

#### NOTE

In practical applications, you can configure OSPF GR on the dual main control boards to avoid service forwarding from being affected by the fault occurred on the main control board.

The AR150/200 can function as only the Helper router, but cannot function as the Restarter router.

#### Pre-configuration Tasks

Before configuring OSPF GR, complete the following tasks:

- Configuring IP addresses for interfaces to ensure that neighboring routers are reachable at the network layer
- [Configuring Basic OSPF Functions](#)

#### Data Preparation

To configure OSPF GR, you need the following data.

| No. | Data                                                                                                               |
|-----|--------------------------------------------------------------------------------------------------------------------|
| 1   | OSPF process number                                                                                                |
| 2   | (Optional) Parameters for establishing GR sessions<br><b>NOTE</b><br>The default parameter values are recommended. |

### 5.10.2 Enabling OSPF GR

Enabling OSPF GR to ensure the fast convergence of OSPF and the stability the network topology.

## Procedure

**Step 1** Run:

```
system-view
```

The system view is displayed.

**Step 2** Run:

```
ospf [process-id]
```

The OSPF view is displayed.

**Step 3** Run:

```
opaque-capability enable
```

The opaque-LSA function is enabled.

The opaque-LSA feature of OSPF needs to be enabled first because OSPF supports GR through Type 9 LSAs.

**Step 4** Run:

```
graceful-restart
```

The OSPF GR feature is enabled.

----End

## 5.10.3 (Optional) Configuring the GR Session Parameters on the Restarter

This part describes how to set GR session parameters (including GR period, planned GR, and totally GR) on the Restarter.

### Procedure

**Step 1** Run:

```
system-view
```

The system view is displayed.

**Step 2** Run:

```
ospf [process-id]
```

The OSPF view is displayed.

**Step 3** Run:

```
graceful-restart [period period | planned-only | partial] *
```

The GR session parameters is set.

- Set **period**, the GR period on the Restarter is set. By default, the restart time is 120 seconds.
- Set **planned-only**, the Restarter supports only the planned GR. By default, the Restarter supports both the planned GR and unplanned GR.
- Set **partial**, the Restarter supports the partial GR. By default, the Restarter supports the totally GR.

----End

## 5.10.4 (Optional) Configuring GR Session Parameters on the Helper

This part describes how to set GR session parameters (including the filtering policies, checks the LSAs outside the AS, and Planned GR) on the Helper.

### Procedure

**Step 1** Run:

```
system-view
```

The system view is displayed.

**Step 2** Run:

```
ospf [process-id]
```

The OSPF view is displayed.

**Step 3** Run:

```
graceful-restart helper-role { [{ ip-prefix ip-prefix-name | acl-number acl-number | acl-name acl-name } | ignore-external-lsa | planned-only] * | never }
```

The GR session parameters is set.

- Set ACL parameters, the local router can enter the Helper mode only after neighbors pass the filtering policies of **ip-prefix** or **acl**.
- Set **ignore-external-lsa**, the Helper does not check the LSAs outside the AS (AS-external LSA). By default, the Helper checks the LSAs outside the AS.
- Set **planned-only**, the Helper supports only the planned GR. By default, the Helper supports both the planned GR and unplanned GR.
- Set **never**, the router does not support the Helper mode.

----End

## 5.10.5 Checking the Configuration

After OSPF GR is configured, you can check the OSPF GR status.

### Prerequisites

The configurations for the OSPF GR are complete.

### Procedure

- Run the **display ospf [ process-id ] graceful-restart [ verbose ]** command to check the restart status of OSPF GR.

----End

## 5.11 Improving Security of an OSPF Network

On a network demanding high security, you can adopt the GTSM mechanism and configure OSPF authentication to improve the security of the OSPF network.

## 5.11.1 Establishing the Configuration Task

Before improving the security of an OSPF network, familiarize yourself with the applicable environment, complete pre-configuration tasks, and obtain the required data. This can help you complete the configuration task quickly and accurately.

### Applicable Environment

In a network demanding high security, you can configure OSPF authentication and adopt the GTSM mechanism to improve the security of the OSPF network.

The GTSM mechanism defends against attacks by checking the TTL value. If an attacker keeps sending packets to a router by simulating real OSPF unicast packets, the router finds itself is the destination of the packets after the interface board receives these packets. The router directly sends the packets to the control plane for OSPF processing without checking the validity of the packets. The router busies itself with processing these "valid" packets. As a result, the system is busy, and the CPU is highly occupied.

The GTSM mechanism protects a router by checking whether the TTL value in the IP packet header is in a pre-defined range to enhance the system security.

#### NOTE

GTSM supports only unicast addresses; therefore, in OSPF, GTSM takes effect on the virtual link and the sham link.

### Pre-configuration Tasks

Before improving the security of an OSPF network, complete the following tasks:

- Configuring IP addresses for interfaces to make neighboring nodes reachable
- [Configuring Basic OSPF Functions](#)

### Data Preparation

To improve the security of an OSPF network, you need the following data.

| No. | Data                                                                        |
|-----|-----------------------------------------------------------------------------|
| 1   | OSPF process ID                                                             |
| 2   | (Optional) Names of VPN instances of OSPF                                   |
| 3   | (Optional) TTL value to be checked                                          |
| 4   | ID of an OSPF area that needs to be configured with authentication          |
| 5   | Number of an OSPF interface that needs to be configured with authentication |
| 6   | Authentication mode and password                                            |

## 5.11.2 Configuring the OSPF GTSM Functions

The GTSM defends against attacks by checking the TTL value.

## Context

To apply GTSM functions, enable GTSM on the two ends of the OSPF connection.

The valid TTL range of the detected packets is  $[255 - hops + 1, 255]$ .

GTSM checks the TTL value of only the packet that matches the GTSM policy. For the packets that do not match the GTSM policy, you can set them as "pass" or "drop". If the GTSM default action performed on the packet is set as "drop", you need to configure all the router connections for GTSM. If the packets sent from a router do not match the GTSM policy, they are dropped. The connection thus cannot be established. This ensures security but reduces the ease of use.

You can enable the log function to record the information that the packets are dropped. This is convenient for fault location.

## Procedure

### Step 1 Run:

```
system-view
```

The system view is displayed.

### Step 2 Run:

```
ospf valid-ttl-hops hops [vpn-instance vpn-instance-name]
```

OSPF GTSM functions are configured.

#### NOTE

The **ospf valid-ttl-hops** command has two functions:

- Enabling OSPF GTSM
- Configuring the TTL value to be detected

The parameter **vpn-instance** is valid only for the latter function.

Thus, if the private network policy or the public network policy is configured only, it is recommended to set the default action performed on the packets that do not match the GTSM policy as **pass**. This prevents the OSPF packets of other processes from being discarded incorrectly.

### Step 3 (Optional) Run:

```
gtsm default-action { drop | pass }
```

The default action performed on the packets that do not match the GTSM policy is set.

By default, the packets that do not match the GTSM policy can pass the filtering.

#### NOTE

If the default action is configured but the GTSM policy is not configured, GTSM does not take effect.

### Step 4 (Optional) Run:

```
gtsm log drop-packet all
```

The log function is enabled on the specified board in the system view. The information that GTSM drops packets is recorded in the log.

----End



## 5.11.3 Configuring the Authentication Mode

OSPF supports packet authentication. Only the packets that pass the authentication can be received. If packets fail to pass the authentication, the neighbor relationship cannot be established.

### Context

In area authentication, all the routers in an area must use the same area authentication mode and password. For example, the authentication mode of all devices in Area 0 is simple authentication and the password is abc.

The interface authentication mode is used among neighbor routers to set the authentication mode and password. Its priority is higher than that of the area authentication mode.

### Procedure

- Configuring the Area Authentication Mode

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
ospf [process-id]
```

The OSPF process view is displayed.

3. Run:

```
area area-id
```

The OSPF area view is displayed.

4. Run the following commands to configure the authentication mode of the OSPF area as required:

- Run:

```
authentication-mode simple [[plain] plain-text | cipher cipher-text]
```

The simple authentication is configured for the OSPF area.

- Run:

```
authentication-mode { md5 | hmac-md5 } [key-id { plain plain-text | [cipher] cipher-text }]
```

The MD5 authentication is configured for the OSPF area.

OSPF supports packet authentication. Only the OSPF packets passing the authentication can be received; otherwise, the neighbor relationship cannot be established normally.

All the routers in an area must agree on the same area authentication mode and password. For example, the authentication mode of all routers in area 0 is simple authentication, and the password is abc.

- Run:

```
authentication-mode keychain keychain-name
```

The Keychain authentication is configured for the OSPF area.

 **NOTE**

Before using the Keychain authentication, you need to configure Keychain information in the system view. To establish the OSPF neighbor relationship, you need to ensure that the **key-id**, **algorithm**, and **key-string** of the local ActiveSendKey are the same as those of the remote ActiveRecvKey.

● **Configuring the Interface Authentication Mode**

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
interface interface-type interface-number
```

The interface view is displayed.

3. Run the following commands to configure the interface authentication mode as required:

- Run:

```
ospf authentication-mode simple [[plain] plain-text | cipher cipher-text]
```

The simple authentication is configured for the OSPF interface.

- Run:

```
ospf authentication-mode { md5 | hmac-md5 } [key-id { plain plain-text | [cipher] cipher-text }]
```

The MD5 authentication is configured for the OSPF interface.

- Run:

```
ospf authentication-mode null
```

The non-authentication mode is configured for the OSPF interface.

- Run:

```
ospf authentication-mode keychain keychain-name
```

The Keychain authentication is configured for the OSPF area.

 **NOTE**

Before using the Keychain authentication, you need to configure Keychain information in the system view. To establish the OSPF neighbor relationship, you need to ensure that the **key-id**, **algorithm**, and **key-string** of the local ActiveSendKey are the same as those of the remote ActiveRecvKey.

The authentication mode and password of interfaces in the same network segment must be consistent except the Keychain authentication mode. If the interfaces are in different network segments, the authentication mode and password of the interfaces can be different.

----End

## 5.11.4 Checking the Configuration

After OSPF features are configured to improve the stability of an OSPF network, you can check GTSM statistics and brief statistics.

### Prerequisites

The configurations for Improving Security of an OSPF Network are complete.

## Procedure

- Run the **display gtsm statistics all** command to check the GTSM statistics.
- Run the **display ospf [ process-id ] request-queue [ interface-type interface-number ] [ neighbor-id ]** command to check the OSPF request queue.
- Run the **display ospf [ process-id ] retrans-queue [ interface-type interface-number ] [ neighbor-id ]** command to check the OSPF retransmission queue.
- Run the **display ospf [ process-id ] error [ lsa ]** or **display ospf error [ packet [ number ] ]** command to check the OSPF error information.

----End

## 5.12 Configuring the Network Management Function of OSPF

OSPF supports the network management function. You can bind the OSPF MIB to a certain OSPF process, and configure the trap function and log function.

### 5.12.1 Establishing the Configuration Task

Before configuring the network management function for OSPF, familiarize yourself with the applicable environment, complete pre-configuration tasks, and obtain the required data. This can help you complete the configuration task quickly and accurately.

#### Applicable Environment

OSPF supports the network management function. You can bind OSPF MIB and a certain OSPF process. In addition, OSPF also supports the trap function and the log function.

#### Pre-configuration Tasks

Before configuring the network management function of OSPF, complete the following tasks:

- Configuring IP addresses for interfaces to make neighboring nodes reachable
- [Configuring Basic OSPF Functions](#)

#### Data Preparation

To configuring the network management function of OSPF, you need the following data.

| No. | Data            |
|-----|-----------------|
| 1   | OSPF process ID |

### 5.12.2 Configuring OSPF MIB Binding

The MIB is a virtual database of the device status maintained by the managed devices.

## Context

When multiple OSPF processes are enabled, you can configure OSPF MIB to select the process to be processed, that is, configure OSPF MIB to select the process to which it is bound.

Do as follows on the OSPF router:

## Procedure

### Step 1 Run:

```
system-view
```

The system view is displayed.

### Step 2 Run:

```
ospf mib-binding process-id
```

OSPF MIB binding is configured.

----End

## 5.12.3 Configuring OSPF Trap

Traps are the notifications sent from a router to inform the NMS of the fault detected by the system.

## Context

Do as follows on the OSPF router.

## Procedure

### Step 1 Run:

```
system-view
```

The system view is displayed.

### Step 2 Run:

```
snmp-agent trap enable feature-name ospf [trap-name { ospfifauthfailure |
ospfifconfigerror | ospfifrxbadpacket | ospfifstatechange |
ospflsdbapproachingoverflow | ospflsdboverflow | ospfmaxagelsa |
ospfnbrrestarthelperstatuschange | ospfnbrstatechange |
ospfnssatranslatorstatuschange | ospforiginatelsa | ospfrestartstatuschange |
ospftxretransmit | ospfvirtifauthfailure | ospfvirtifconfigerror |
ospfvirtifrxbadpacket | ospfvirtifstatechange | ospfvirtiftxretransmit |
ospfvirtnbrrestarthelperstatuschange | ospfvirtnbrstatechange }]
```

The trap function for the OSPF module is enabled.

To enable the traps of one or more events, you can specify **type-name**.

----End

## 5.12.4 Configuring OSPF Log

Logs record the operations (such as configuring commands) and specific events (such as the network connection failure) on routers.

## Context

Do as follows on the OSPF router:

## Procedure

### Step 1 Run:

```
system-view
```

The system view is displayed.

### Step 2 Run:

```
ospf [process-id]
```

The OSPF process view is displayed.

### Step 3 Run:

```
enable log [config | error | state | snmp-trap]
```

The log function is enabled.

----End

## 5.12.5 Checking the Configuration

After the network management function is configured for OSPF, you can check the contents of the information channel, information recorded in the information center, log buffer, and trap buffer.

## Prerequisites

The configurations for the network management function of OSPF are complete.

## Procedure

- Run the **display ospf [ process-id ] brief** command to view information about the binding of OSPF MIBs and OSPF processes.
- Run the **display snmp-agent trap feature-name ospf all** command to view all trap messages of the OSPF module.

----End

## 5.13 Maintaining OSPF

Maintaining OSPF involves resetting OSPF and clearing OSPF statistics.

### 5.13.1 Resetting OSPF

Restarting OSPF can reset OSPF. In addition, you can reset OSPF through GR.

## Context



The OSPF neighbor relationship is deleted after you reset OSPF connections with the **reset ospf** command. Exercise caution when running this command.

---

To reset OSPF connections, run the following **reset ospf** commands in the user view.

## Procedure

- Run the **reset ospf [ process-id ] process [ flush-waiting-timer time ]** command in the user view to Restart the OSPF process.
- Run the **reset ospf [ process-id ] process [ graceful-restart ]** command in the user view to Restart the OSPF process in GR mode.

----End

### 5.13.2 Clearing OSPF

This section describes how to clear OSPF statistics, including OSPF counters, imported routes, and GTSM statistics on the board.

## Context



OSPF information cannot be restored after being cleared. Exercise caution when running this command.

---

To clear the OSPF information, run the following **reset ospf** commands in the user view.

## Procedure

- Run the **reset ospf [ process-id ] counters [ neighbor [ interface-type interface-number ] [ router-id ] ]** command in the user view to clear OSPF counters.
- Run the **reset ospf [ process-id ] redistribution** command in the user view to clear the routes imported by OSPF.
- Run the **reset gtsm statistics all** command in the user view to clear the GTSM statistics on the board.

----End

## 5.14 Configuration Examples

This section provides several configuration examples of OSPF together with the configuration flowchart. The configuration examples explain networking requirements, configuration notes, and configuration roadmap.

## 5.14.1 Example for Configuring Basic OSPF Functions

This part provides an example for configuring basic OSPF functions. Detailed operations include enabling OSPF on each router and specifying network segments in different areas.

### Networking Requirements

As shown in [Figure 5-5](#), all routers run OSPF, and the entire AS is divided into three areas. Router A and Router B serve as ABRs to forward routes between areas.

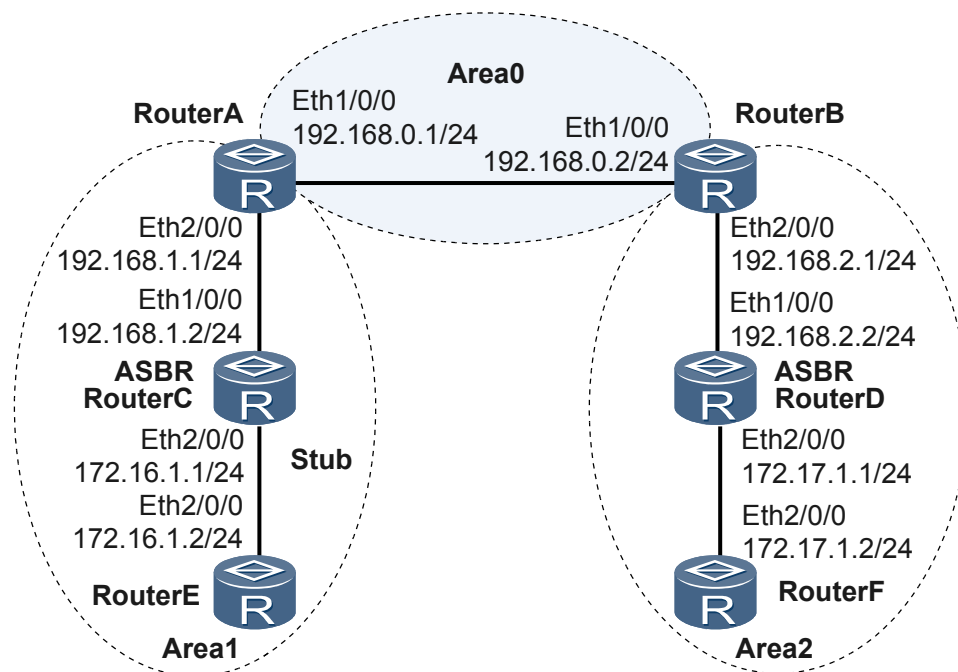
After the configuration, each router should learn the routes from the AS to all network segments.



**NOTE**

AR150/200 is RouterE or RouterF.

**Figure 5-5** Networking diagram of configuring basic OSPF functions



### Configuration Roadmap

The configuration roadmap is as follows:

1. Enable OSPF on each router.
2. Specify network segments in different areas.

### Data Preparation

To complete the configuration, you need the following data:

- The router ID of Router A is 1.1.1.1, the OSPF process number is 1, the network segment of Area 0 is 192.168.0.0/24, and the network segment of Area 1 is 192.168.1.0/24.

- The router ID of Router B is 2.2.2.2, the OSPF process number is 1, the network segment of Area 0 is 192.168.0.0/24, and the network segment of Area 2 is 192.168.2.0/24.
- The router ID of Router C is 3.3.3.3, the OSPF process number is 1, and the network segments of Area 1 are 192.168.1.0/24 and 172.16.1.0/24.
- The router ID of Router D is 4.4.4.4, the OSPF process number is 1, and the network segments of Area 2 are 192.168.2.0/24 and 172.17.1.0/24.
- The router ID of Router E is 5.5.5.5, the OSPF process number is 1, and the network segment of Area 1 is 172.16.1.0/24.
- The router ID of Router F is 6.6.6.6, the OSPF process number is 1, and the network segment of Area 2 is 172.17.1.0/24.

## Procedure

### Step 1 Configure an IP address for each interface.

The configuration details are not mentioned here.

### Step 2 Configure basic OSPF functions.

# Configure Router A.

```
[RouterA] router id 1.1.1.1
[RouterA] ospf
[RouterA-ospf-1] area 0
[RouterA-ospf-1-area-0.0.0.0] network 192.168.0.0 0.0.0.255
[RouterA-ospf-1-area-0.0.0.0] quit
[RouterA-ospf-1] area 1
[RouterA-ospf-1-area-0.0.0.1] network 192.168.1.0 0.0.0.255
[RouterA-ospf-1-area-0.0.0.1] quit
```

# Configure Router B.

```
[RouterB] router id 2.2.2.2
[RouterB] ospf
[RouterB-ospf-1] area 0
[RouterB-ospf-1-area-0.0.0.0] network 192.168.0.0 0.0.0.255
[RouterB-ospf-1-area-0.0.0.0] quit
[RouterB-ospf-1] area 2
[RouterB-ospf-1-area-0.0.0.2] network 192.168.2.0 0.0.0.255
[RouterB-ospf-1-area-0.0.0.2] quit
```

# Configure Router C.

```
[RouterC] router id 3.3.3.3
[RouterC] ospf
[RouterC-ospf-1] area 1
[RouterC-ospf-1-area-0.0.0.1] network 192.168.1.0 0.0.0.255
[RouterC-ospf-1-area-0.0.0.1] network 172.16.1.0 0.0.0.255
[RouterC-ospf-1-area-0.0.0.1] quit
```

# Configure Router D.

```
[RouterD] router id 4.4.4.4
[RouterD] ospf
[RouterD-ospf-1] area 2
[RouterD-ospf-1-area-0.0.0.2] network 192.168.2.0 0.0.0.255
[RouterD-ospf-1-area-0.0.0.2] network 172.17.1.0 0.0.0.255
[RouterD-ospf-1-area-0.0.0.2] quit
```

# Configure Router E.

```
[RouterE] router id 5.5.5.5
[RouterE] ospf
[RouterE-ospf-1] area 1
```



```
[RouterE-ospf-1-area-0.0.0.1] network 172.16.1.0 0.0.0.255
[RouterE-ospf-1-area-0.0.0.1] quit
```

## # Configure Router F.

```
[RouterF] router id 6.6.6.6
[RouterF] ospf
[RouterF-ospf-1] area 2
[RouterF-ospf-1-area-0.0.0.2] network 172.17.1.0 0.0.0.255
[RouterF-ospf-1-area-0.0.0.2] quit
```

**Step 3** Verify the configuration.

## # View OSPF neighbors of Router A.

```
[RouterA] display ospf peer
OSPF Process 1 with Router ID 1.1.1.1
Neighbors
Area 0.0.0.0 interface 192.168.0.1(Ethernet1/0/0)'s neighbors
Router ID: 2.2.2.2 Address: 192.168.0.2
State: Full Mode:Nbr is Master Priority: 1
DR: 192.168.0.2 BDR: 192.168.0.1 MTU: 0
Dead timer due in 36 sec
Retrans timer interval: 5
Neighbor is up for 00:15:04
Authentication Sequence: [0]
Neighbors
Area 0.0.0.1 interface 192.168.1.1(Ethernet2/0/0)'s neighbors
Router ID: 3.3.3.3 Address: 192.168.1.2
State: Full Mode:Nbr is Master Priority: 1
DR: 192.168.1.2 BDR: 192.168.1.1 MTU: 0
Dead timer due in 39 sec
Retrans timer interval: 5
Neighbor is up for 00:07:32
Authentication Sequence: [0]
```

## # View the OSPF routing information of Router A.

```
[RouterA] display ospf routing
OSPF Process 1 with Router ID 1.1.1.1
Routing Tables
Routing for Network
Destination Cost Type NextHop AdvRouter Area
172.16.1.0/24 2 Transit 192.168.1.2 3.3.3.3 0.0.0.1
172.17.1.0/24 3 Inter-area 192.168.0.2 2.2.2.2 0.0.0.0
192.168.0.0/24 1 Stub 192.168.0.1 1.1.1.1 0.0.0.0
192.168.1.0/24 1 Stub 192.168.1.1 1.1.1.1 0.0.0.1
192.168.2.0/24 2 Inter-area 192.168.0.2 2.2.2.2 0.0.0.0
Total Nets: 5
Intra Area: 3 Inter Area: 2 ASE: 0 NSSA: 0
```

## # View the LSDB of Router A.

```
[RouterA] display ospf lsdb
OSPF Process 1 with Router ID 1.1.1.1
Link State Database
Area: 0.0.0.0
Type LinkState ID AdvRouter Age Len Sequence Metric
Router 2.2.2.2 2.2.2.2 317 48 80000003 1
Router 1.1.1.1 1.1.1.1 316 48 80000002 1
Sum-Net 172.16.1.0 1.1.1.1 250 28 80000001 2
Sum-Net 172.17.1.0 2.2.2.2 203 28 80000001 2
Sum-Net 192.168.2.0 2.2.2.2 237 28 80000002 1
Sum-Net 192.168.1.0 1.1.1.1 295 28 80000002 1
Area: 0.0.0.1
Type LinkState ID AdvRouter Age Len Sequence Metric
Router 5.5.5.5 5.5.5.5 214 36 80000004 1
Router 3.3.3.3 3.3.3.3 217 60 80000008 1
Router 1.1.1.1 1.1.1.1 289 48 80000002 1
Network 172.16.1.1 3.3.3.3 670 32 80000001 0
```

```
Sum-Net 172.17.1.0 1.1.1.1 202 28 80000001 3
Sum-Net 192.168.2.0 1.1.1.1 242 28 80000001 2
Sum-Net 192.168.0.0 1.1.1.1 300 28 80000001 1
```

# View the routing table of Router D and test connectivity by using the **ping** command.

```
[RouterD] display ospf routing
 OSPF Process 1 with Router ID 4.4.4.4
 Routing Tables
Routing for Network
Destination Cost Type NextHop AdvRouter Area
172.16.1.0/24 4 Inter-area 192.168.2.1 2.2.2.2 0.0.0.2
172.17.1.0/24 1 Transit 172.17.1.1 4.4.4.4 0.0.0.2
192.168.0.0/24 2 Inter-area 192.168.2.1 2.2.2.2 0.0.0.2
192.168.1.0/24 3 Inter-area 192.168.2.1 2.2.2.2 0.0.0.2
192.168.2.0/24 1 Stub 192.168.2.2 4.4.4.4 0.0.0.2
Total Nets: 5
Intra Area: 2 Inter Area: 3 ASE: 0 NSSA: 0
[RouterD] ping 172.16.1.1
PING 172.16.1.1: 56 data bytes, press CTRL_C to break
 Reply from 172.16.1.1: bytes=56 Sequence=1 ttl=253 time=62 ms
 Reply from 172.16.1.1: bytes=56 Sequence=2 ttl=253 time=16 ms
 Reply from 172.16.1.1: bytes=56 Sequence=3 ttl=253 time=62 ms
 Reply from 172.16.1.1: bytes=56 Sequence=4 ttl=253 time=94 ms
 Reply from 172.16.1.1: bytes=56 Sequence=5 ttl=253 time=63 ms
--- 172.16.1.1 ping statistics ---
 5 packet(s) transmitted
 5 packet(s) received
 0.00% packet loss
 round-trip min/avg/max = 16/59/94 ms
```

---End

## Configuration Files

- Configuration file of Router A

```
#
 sysname RouterA
#
interface Ethernet1/0/0
 ip address 192.168.0.1 255.255.255.0
#
interface Ethernet2/0/0
 ip address 192.168.1.1 255.255.255.0
#
ospf 1
 area 0.0.0.0
 network 192.168.0.0 0.0.0.255
 area 0.0.0.1
 network 192.168.1.0 0.0.0.255
#
return
```

- Configuration file of Router B

```
#
 sysname RouterB
#
router id 2.2.2.2
#
interface Ethernet1/0/0
 ip address 192.168.0.2 255.255.255.0
#
interface Ethernet2/0/0
 ip address 192.168.2.1 255.255.255.0
#
ospf 1
 area 0.0.0.0
 network 192.168.0.0 0.0.0.255
```

```
 area 0.0.0.2
 network 192.168.2.0 0.0.0.255
#
return
```

- Configuration file of Router C

```
#
sysname RouterC
#
router id 3.3.3.3
#
interface Ethernet1/0/0
 ip address 192.168.1.2 255.255.255.0
#
interface Ethernet2/0/0
 ip address 172.16.1.1 255.255.255.0
#
ospf 1
 area 0.0.0.1
 network 192.168.1.0 0.0.0.255
 network 172.16.1.0 0.0.0.255
#
return
```

- Configuration file of Router D

```
#
sysname RouterD
#
router id 4.4.4.4
#
interface Ethernet1/0/0
 ip address 192.168.2.2 255.255.255.0
#
interface Ethernet2/0/0
 ip address 172.17.1.1 255.255.255.0
#
ospf 1
 area 0.0.0.2
 network 192.168.2.0 0.0.0.255
 network 172.17.1.0 0.0.0.255
#
return
```

- Configuration file of Router E

```
#
sysname RouterE
#
router id 5.5.5.5
#
interface Ethernet2/0/0
 ip address 172.16.1.2 255.255.255.0
#
ospf 1
 area 0.0.0.1
 network 172.16.1.0 0.0.0.255
#
return
```

- Configuration file of Router F

```
#
sysname RouterF
#
router id 6.6.6.6
#
interface Ethernet2/0/0
 ip address 172.17.1.2 255.255.255.0
#
ospf 1
 area 0.0.0.2
```

```
network 172.17.1.0 0.0.0.255
#
return
```

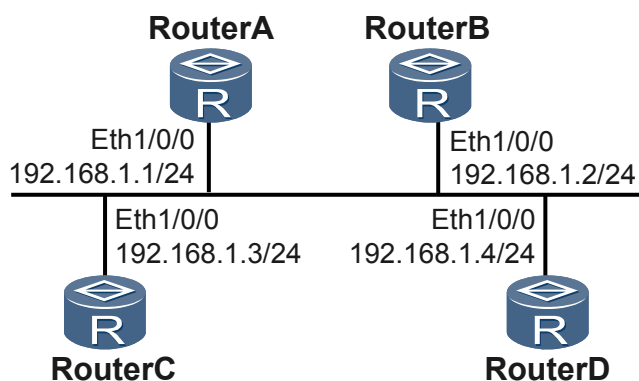
## 5.14.2 Example for Configuring DR Election of OSPF

This part provides an example for setting the DR priority on an interface for DR election on a broadcast network.

### Networking Requirements

As shown in [Figure 5-6](#), Router A has the highest priority (100) in the network and is elected as the DR. Router C has the second highest priority, and is elected as the BDR. The priority of Router B is 0, and Router B cannot be elected as the DR or BDR. The priority of Router D is not configured and its default value is 1.

Figure 5-6 Configuring DR election of OSPF



### Configuration Roadmap

The configuration roadmap is as follows:

1. Configure the router ID on each router, enable OSPF, and specify the network segment.
2. Check the DR/BDR status of each router with the default priority.
3. Configure the DR priority of the interface and check the DR/BDR status.

### Data Preparation

To complete the configuration, you need the following data:

- The router ID of Router A is 1.1.1.1 and the DR priority is 100.
- The router ID of Router B is 2.2.2.2 and the DR priority is 0.
- The router ID of Router C is 3.3.3.3 and the DR priority is 2.
- The router ID of Router D is 4.4.4.4 and the DR priority is 1.

### Procedure

**Step 1** Configure an IP address for each interface.

The configuration details are not mentioned here.

## Step 2 Configure basic OSPF functions.

### # Configure Router A.

```
[RouterA] router id 1.1.1.1
[RouterA] ospf
[RouterA-ospf-1] area 0
[RouterA-ospf-1-area-0.0.0.0] network 192.168.1.0 0.0.0.255
[RouterA-ospf-1-area-0.0.0.0] quit
```

### # Configure Router B.

```
[RouterB] router id 2.2.2.2
[RouterB] ospf
[RouterB-ospf-1] area 0
[RouterB-ospf-1-area-0.0.0.0] network 192.168.1.0 0.0.0.255
[RouterB-ospf-1-area-0.0.0.0] quit
```

### # Configure Router C.

```
[RouterC] router id 3.3.3.3
[RouterC] ospf
[RouterC-ospf-1] area 0
[RouterC-ospf-1-area-0.0.0.0] network 192.168.1.0 0.0.0.255
[RouterC-ospf-1-area-0.0.0.0] quit
```

### # Configure Router D.

```
[RouterD] router id 4.4.4.4
[RouterD] ospf
[RouterD-ospf-1] area 0
[RouterD-ospf-1-area-0.0.0.0] network 192.168.1.0 0.0.0.255
[RouterD-ospf-1-area-0.0.0.0] quit
```

### # View the DR/BDR status.

```
[RouterA] display ospf peer
 OSPF Process 1 with Router ID 1.1.1.1
 Neighbors
 Area 0.0.0.0 interface 192.168.1.1(Ethernet1/0/0)'s neighbors
 Router ID: 2.2.2.2 Address: 192.168.1.2
State: Full Mode:Nbr is Master Priority: 1
DR: 192.168.1.4 BDR: 192.168.1.3 MTU: 0
 Dead timer due in 32 sec
 Retrans timer interval: 5
 Neighbor is up for 00:04:21
 Authentication Sequence: [0]
 Router ID: 3.3.3.3 Address: 192.168.1.3
State: Full Mode:Nbr is Master Priority: 1
DR: 192.168.1.4 BDR: 192.168.1.3 MTU: 0
 Dead timer due in 37 sec
 Retrans timer interval: 5
 Neighbor is up for 00:04:06
 Authentication Sequence: [0]
 Router ID: 4.4.4.4 Address: 192.168.1.4
State: Full Mode:Nbr is Master Priority: 1
DR: 192.168.1.4 BDR: 192.168.1.3 MTU: 0
 Dead timer due in 37 sec
 Retrans timer interval: 5
 Neighbor is up for 00:03:53
 Authentication Sequence: [0]
```

# View the neighbor information of Router A. You can see the priority of DR and the neighbor status. The Router D is the DR, and Router C is the BDR.

 **NOTE**

When the priority is the same, the router with a higher router ID is elected as the DR. If a new router is added after the DR/BDR election is complete, the new router cannot become the DR even if it has the highest priority.

**Step 3** Configure DR priorities on interfaces.

## # Configure Router A.

```
[RouterA] interface Ethernet 1/0/0
[RouterA-Ethernet1/0/0] ospf dr-priority 100
[RouterA-Ethernet1/0/0] quit
```

## # Configure Router B.

```
[RouterB] interface Ethernet 1/0/0
[RouterB-Ethernet1/0/0] ospf dr-priority 0
[RouterB-Ethernet1/0/0] quit
```

## # Configure Router C.

```
[RouterC] interface Ethernet 1/0/0
[RouterC-Ethernet1/0/0] ospf dr-priority 2
[RouterC-Ethernet1/0/0] quit
```

## # View the DR/BDR status.

```
[RouterD] display ospf peer
 OSPF Process 1 with Router ID 4.4.4.4
 Neighbors
Area 0.0.0.0 interface 192.168.1.4(Ethernet1/0/0)'s neighbors
Router ID: 1.1.1.1 Address: 192.168.1.1
State: Full Mode:Nbr is Slave Priority: 100
DR: 192.168.1.4 BDR: 192.168.1.3 MTU: 0
Dead timer due in 31 sec
Retrans timer interval: 5
Neighbor is up for 00:11:17
Authentication Sequence: [0]
Router ID: 2.2.2.2 Address: 192.168.1.2
State: Full Mode:Nbr is Slave Priority: 0
DR: 192.168.1.4 BDR: 192.168.1.3 MTU: 0
Dead timer due in 35 sec
Retrans timer interval: 5
Neighbor is up for 00:11:19
Authentication Sequence: [0]
Router ID: 3.3.3.3 Address: 192.168.1.3
State: Full Mode:Nbr is Slave Priority: 2
DR: 192.168.1.4 BDR: 192.168.1.3 MTU: 0
Dead timer due in 33 sec
Retrans timer interval: 5
Neighbor is up for 00:11:15
Authentication Sequence: [0]
```

**Step 4** Restart OSPF processes.

In the user view of each router, run the **reset ospf 1 process** command to restart the OSPF process.

**Step 5** View the configuration.

## # View the status of OSPF neighbors.

```
[RouterD] display ospf peer
 OSPF Process 1 with Router ID 4.4.4.4
 Neighbors
Area 0.0.0.0 interface 192.168.1.4(Ethernet1/0/0)'s neighbors
Router ID: 1.1.1.1 Address: 192.168.1.1
State: Full Mode:Nbr is Slave Priority: 100
```

```
DR: 192.168.1.1 BDR: 192.168.1.3 MTU: 0
 Dead timer due in 35 sec
 Retrans timer interval: 5
 Neighbor is up for 00:07:19
 Authentication Sequence: [0]
 Router ID: 2.2.2.2 Address: 192.168.1.2
State: Full Mode:Nbr is Master Priority: 0
DR: 192.168.1.1 BDR: 192.168.1.3 MTU: 0
 Dead timer due in 35 sec
 Retrans timer interval: 5
 Neighbor is up for 00:07:19
 Authentication Sequence: [0]
 Router ID: 3.3.3.3 Address: 192.168.1.3
State: Full Mode:Nbr is Slave Priority: 2
DR: 192.168.1.1 BDR: 192.168.1.3 MTU: 0
 Dead timer due in 37 sec
 Retrans timer interval: 5
 Neighbor is up for 00:07:17
 Authentication Sequence: [0]
```

# View the status of the OSPF interface.

```
[RouterA] display ospf interface
 OSPF Process 1 with Router ID 1.1.1.1
 Interfaces
Area: 0.0.0.0
IP Address Type State Cost Pri DR BDR
192.168.1.1 Broadcast DR 1 100 192.168.1.1 192.168.1.3
[RouterB] display ospf interface
 OSPF Process 1 with Router ID 2.2.2.2
 Interfaces
Area: 0.0.0.0
IP Address Type State Cost Pri DR BDR
192.168.1.2 Broadcast DROther 1 0 192.168.1.1 192.168.1.3
```

If all neighbors are in the Full state, it indicates that Router A establishes the neighbor relationship with its neighbor. If the neighbor stays "2-Way", it indicates both of them are not the DR or BDR. They need not exchange LSAs.

If the status of the OSPF interface is DROther, it indicates that it is neither DR nor BDR.

----End

## Configuration Files

- Configuration file of Router A

```
#
sysname RouterA
#
router id 1.1.1.1
#
interface Ethernet1/0/0
ip address 192.168.1.1 255.255.255.0
ospf dr-priority 100
#
ospf 1
area 0.0.0.0
network 192.168.1.0 0.0.0.255
#
return
```
- Configuration file of Router B

```
#
sysname RouterB
#
router id 2.2.2.2
#
```

```
interface Ethernet1/0/0
 ip address 192.168.1.2 255.255.255.0
 ospf dr-priority 0
#
ospf 1
 area 0.0.0.0
 network 192.168.1.0 0.0.0.255
#
return
```

- Configuration file of Router C

```
#
 sysname RouterC
#
router id 3.3.3.3
#
interface Ethernet1/0/0
 ip address 192.168.1.3 255.255.255.0
 ospf dr-priority 2
#
ospf 1
 area 0.0.0.0
 network 192.168.1.0 0.0.0.255
#
return
```

- Configuration file of Router D

```
#
 sysname RouterD
#
router id 4.4.4.4
#
interface Ethernet1/0/0
 ip address 192.168.1.4 255.255.255.0
#
ospf 1
 area 0.0.0.0
 network 192.168.1.0 0.0.0.255
#
return
```

### 5.14.3 Example for Configuring OSPF Stub Areas

This part provides an example for configuring a stub area that imports static routes to reduce the number of LSAs advertised in this area without affecting the route reachability.

#### Networking Requirements

As shown in [Figure 5-7](#), all routers run OSPF, and the entire AS is divided into three areas. Router A and Router B serve as ABRs to forward routes between areas. Router D serves as an ASBR to import external routes (static routes).

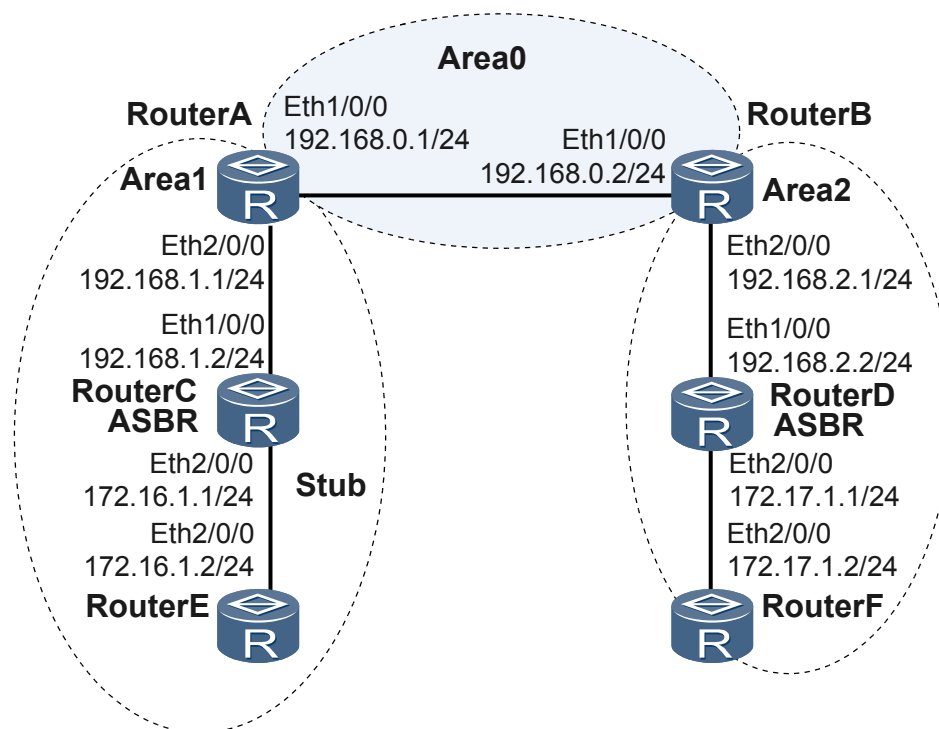
It is required to configure Area 1 as a stub area to reduce the LSAs advertised to this area without affecting the route reachability.

 **NOTE**

AR150/200 is RouterE or RouterF.



Figure 5-7 Configuring OSPF stub areas



## Configuration Roadmap

The configuration roadmap is as follows:

1. Enable OSPF on each router, and configure basic OSPF functions.
2. Configure static routes on Router D, and import them into OSPF.
3. Configure Area 1 as a stub area, and check the OSPF routing information on Router C.
4. Stop Router A from advertising Type 3 LSAs to the stub area, and check the OSPF routing information on Router C.

## Data Preparation

To complete the configuration, you need the following data:

- The router ID of Router A is 1.1.1.1, the OSPF process number is 1, the network segment of Area 0 is 192.168.0.0/24, and the network segment of Area 1 is 192.168.1.0/24.
- The router ID of Router B is 2.2.2.2, the OSPF process number is 1, the network segment of Area 0 is 192.168.0.0/24, and the network segment of Area 2 is 192.168.2.0/24.
- The router ID of Router C is 3.3.3.3, the OSPF process number is 1, and the network segments of Area 1 are 192.168.1.0/24 and 172.16.1.0/24.
- The router ID of Router D is 4.4.4.4, the OSPF process number is 1, and the network segments of Area 2 are 192.168.2.0/24 and 172.17.1.0/24.
- The router ID of Router E is 5.5.5.5, the OSPF process number is 1, and the network segment of Area 1 is 172.16.1.0/24.

- The router ID of Router F is 6.6.6.6, the OSPF process number is 1, and the network segment of Area 2 is 172.17.1.0/24.

## Procedure

**Step 1** Configure an IP address for each interface.

The configuration details are not mentioned here.

**Step 2** Configure basic OSPF functions (see [Example for Configuring Basic OSPF Functions](#)).

**Step 3** Configure Router D to import static routes.

```
[RouterD] ip route-static 200.0.0.0 8 null 0
[RouterD] ospf
[RouterD-ospf-1] import-route static type 1
[RouterD-ospf-1] quit
```

# View ABR/ASBR information on Router C.

```
[RouterC] display ospf abr-asbr
 OSPF Process 1 with Router ID 3.3.3.3
 Routing Table to ABR and ASBR

RtType Destination Area Cost Nexthop Type

Intra-area 1.1.1.1 0.0.0.1 1 192.168.1.1 ABR
Inter-area 4.4.4.4 0.0.0.1 3 192.168.1.1 ASBR
```

# View the OSPF routing table of Router C.

### NOTE

When Router C is in a common area, there are AS external routes in the routing table.

```
[RouterC] display ospf routing
 OSPF Process 1 with Router ID 3.3.3.3
 Routing Tables

Routing for Network
Destination Cost Type NextHop AdvRouter Area

172.16.1.0/24 1 Transit 172.16.1.1 3.3.3.3 0.0.0.1
172.17.1.0/24 4 Inter-area 192.168.1.1 1.1.1.1 0.0.0.1
192.168.0.0/24 2 Inter-area 192.168.1.1 1.1.1.1 0.0.0.1
192.168.1.0/24 1 Stub 192.168.1.2 3.3.3.3 0.0.0.1
192.168.2.0/24 3 Inter-area 192.168.1.1 1.1.1.1 0.0.0.1

Routing for ASEs
Destination Cost Type Tag NextHop AdvRouter

200.0.0.0/8 4 Type1 1 192.168.1.1 4.4.4.4
Total Nets: 6
Intra Area: 2 Inter Area: 3 ASE: 1 NSSA: 0
```

**Step 4** Configure Area 1 as a stub area.

# Configure Router A.

```
[RouterA] ospf
[RouterA-ospf-1] area 1
[RouterA-ospf-1-area-0.0.0.1] stub
[RouterA-ospf-1-area-0.0.0.1] quit
```

# Configure Router C.

```
[RouterC] ospf
[RouterC-ospf-1] area 1
[RouterC-ospf-1-area-0.0.0.1] stub
[RouterC-ospf-1-area-0.0.0.1] quit
```

# Configure Router E.

```
[RouterE] ospf
[RouterE-ospf-1] area 1
```

```
[RouterE-ospf-1-area-0.0.0.1] stub
[RouterE-ospf-1-area-0.0.0.1] quit
```

# View the routing table of Router C.

#### NOTE

After the area where Router C resides is configured as a stub area, AS external routes are invisible. Instead, there is a default route.

```
[RouterC] display ospf routing
 OSPF Process 1 with Router ID 3.3.3.3
 Routing Tables

Routing for Network
Destination Cost Type NextHop AdvRouter Area
0.0.0.0/0 2 Inter-area 192.168.1.1 1.1.1.1 0.0.0.1
172.16.1.0/24 1 Transit 172.16.1.1 3.3.3.3 0.0.0.1
172.17.1.0/24 4 Inter-area 192.168.1.1 1.1.1.1 0.0.0.1
192.168.0.0/24 2 Inter-area 192.168.1.1 1.1.1.1 0.0.0.1
192.168.1.0/24 1 Stub 192.168.1.2 3.3.3.3 0.0.0.1
192.168.2.0/24 3 Inter-area 192.168.1.1 1.1.1.1 0.0.0.1
Total Nets: 6
Intra Area: 2 Inter Area: 4 ASE: 0 NSSA: 0
```

**Step 5** # Stop Router A from advertising Type 3 LSAs to the stub area.

```
[RouterA] ospf
[RouterA-ospf-1] area 1
[RouterA-ospf-1-area-0.0.0.1] stub no-summary
[RouterA-ospf-1-area-0.0.0.1] quit
```

**Step 6** Verify the configuration.

# View the OSPF routing table of Router C.

```
[RouterC] display ospf routing
 OSPF Process 1 with Router ID 3.3.3.3
 Routing Tables

Routing for Network
Destination Cost Type NextHop AdvRouter Area
0.0.0.0/0 2 Inter-area 192.168.1.1 1.1.1.1 0.0.0.1
172.16.1.0/24 1 Transit 172.16.1.1 3.3.3.3 0.0.0.1
192.168.1.0/24 1 Stub 192.168.1.2 3.3.3.3 0.0.0.1
Total Nets: 3
Intra Area: 2 Inter Area: 1 ASE: 0 NSSA: 0
```

#### NOTE

After the advertisement of summary LSAs to a stub area is disabled, the routing entries of the stub router are further reduced, and only the default route to a destination outside the AS is reserved.

----End

## Configuration Files

#### NOTE

The configuration files of Router B and Router F are the same as those in the preceding example, and are not mentioned here.

#### ● Configuration file of Router A

```
#
 sysname RouterA
#
router id 1.1.1.1
#
interface Ethernet1/0/0
 ip address 192.168.0.1 255.255.255.0
#
interface Ethernet2/0/0
```

```
 ip address 192.168.1.1 255.255.255.0
#
ospf 1
 area 0.0.0.0
 network 192.168.0.0 0.0.0.255
 area 0.0.0.1
 network 192.168.1.0 0.0.0.255
 stub no-summary
#
return
```

- Configuration file of Router C

```
#
sysname RouterC
#
router id 3.3.3.3
#
interface Ethernet1/0/0
 ip address 192.168.1.2 255.255.255.0
#
interface Ethernet2/0/0
 ip address 172.16.1.1 255.255.255.0
#
ospf 1
 area 0.0.0.1
 network 192.168.1.0 0.0.0.255
 network 172.16.1.0 0.0.0.255
 stub
#
return
```

- Configuration file of Router D

```
#
sysname RouterD
#
router id 4.4.4.4
#
interface Ethernet1/0/0
 ip address 192.168.2.2 255.255.255.0
#
interface Ethernet2/0/0
 ip address 172.17.1.1 255.255.255.0
#
ospf 1
 import-route static type 1
 area 0.0.0.2
 network 192.168.2.0 0.0.0.255
 network 172.17.1.0 0.0.0.255
#
ip route-static 200.0.0.0 255.0.0.0 NULL0
#
return
```

- Configuration file of Router E

```
#
sysname RouterE
#
router id 5.5.5.5
#
interface Ethernet2/0/0
 ip address 172.16.1.2 255.255.255.0
#
ospf 1
 area 0.0.0.1
 network 172.16.1.0 0.0.0.255
 stub
#
return
```

# 6 OSPFv3 Configuration

---

## About This Chapter

By building Open Shortest Path First Version 3 (OSPFv3) networks, you can enable OSPFv3 to discover and calculate routes in ASs. OSPFv3 is applicable to a large-scale network that consists of hundreds of routers.

### [6.1 OSPFv3 Overview](#)

OSPFv3 uses the same implementation mechanism as OSPFv2 but is not compatible with OSPFv2.

### [6.2 OSPFv3 Features Supported by AR150/200](#)

The AR150/200 supports various OSPFv3 features, including multi-process and GR.

### [6.3 Configuring Basic OSPFv3 Functions](#)

Before building OSPFv3 networks, you need to configure basic OSPFv3 functions.

### [6.4 Establishing or Maintaining OSPFv3 Neighbor Relationship](#)

By establishing and maintaining OSPFv3 neighbor relationships or adjacencies, you can build OSPFv3 networks.

### [6.5 Configuring OSPFv3 Areas](#)

OSPFv3 supports stub areas and virtual links, the principle and applicable environment of which are similar to those in OSPFv2.

### [6.6 Configuring OSPFv3 NSSA Areas](#)

By configuring areas as NSSA areas, external routes can be imported, and a new type of LSA, namely, Type 7 NSSA LSA is introduced.

### [6.7 Configuring OSPFv3 Route Attributes](#)

By setting OSPFv3 route attributes, you can change OSPFv3 routing policies to meet the requirements of complex networks.

### [6.8 Controlling OSPFv3 Routing Information](#)

This section describes how to control OSPFv3 routing information. Detailed operations include configuring route aggregation, filtering the received routes, and importing external routes.

### [6.9 Optimizing an OSPFv3 Network](#)

By configuring OSPFv3 functions in special network environments, you can adjust and optimize the OSPFv3 network performance.

### 6.10 Configuration OSPFv3 GR

By configuring OSPFv3 GR, you can avoid inaccurate route calculation and packet loss after an OSPFv3 router restarts.

### 6.11 Configuring the Network Management Function of OSPFv3

OSPFv3 supports the network management function. You can bind the OSPFv3 MIB to a certain OSPFv3 process.

### 6.12 Maintaining OSPFv3

Maintaining OSPFv3 and Debugging OSPFv3 involve resetting OSPFv3.

### 6.13 Configuration Examples

This section provides several configuration examples of OSPFv3 together with the configuration flowchart. The configuration examples explain networking requirements, configuration notes, and configuration roadmap.

## 6.1 OSPFv3 Overview

OSPFv3 uses the same implementation mechanism as OSPFv2 but is not compatible with OSPFv2.

The Open Shortest Path First Version 3.0 (OSPFv3) supports the version 6 of the Internet Protocol (IPv6). OSPFv3 conforms to RFC 2740 (OSPF for IPv6).

OSPFv3 and OSPFv2 have the following in common:

- 32-bit Router ID, Area ID, and Link State Advertisement (LSA) link-state ID
- Five types of packets such as Hello, Database Description (DD), Link State Request (LSR), Link State Update (LSU), and Link State Acknowledgement (LSAck) packets
- Neighbor discovery and adjacency establishment mechanisms
- Flooding and aging mechanisms of LSAs
- LSA types

OSPFv3 and OSPFv2 differ as follows:

- OSPFv3 runs based on a link; OSPFv2 runs based on a network segment.
- OSPFv3 can run multiple instances on the same link.
- The topology of OSPFv3 is independent of IPv6 address prefixes.
- OSPFv3 identifies its neighbors with the IPv6 link-local addresses.
- OSPFv3 has three new types of LSA flooding scopes.

## 6.2 OSPFv3 Features Supported by AR150/200

The AR150/200 supports various OSPFv3 features, including multi-process and GR.

The AR150/200 supports the following OSPFv3 features:

- Basic features stipulated in RFC 2740
- OSPFv3 stub areas
- OSPFv3 multi-process
- Multiple OSPFv3 processes can run on a router.
- OSPFv3 GR
  - If a router restarts or performs the active/standby switchover, it directly ages all the entries in the Forward Information Base (FIB). This interrupts the routing. The neighboring routers remove the router from the neighbor list and inform other routers of the router failure. Then, SPF needs to be calculated again. If the router recovers after a short period of time, the neighbor relationship becomes unstable. This results in route flapping.
  - If a router restarts because of abnormalities, you can enable OSPFv3 Graceful Restart (GR) to avoid service interruption during the restart of the router.

 **NOTE**

The OSPFv3 function is used with a license. To use the OSPFv3 function, apply for and purchase the following license from the Huawei local office:

- AR150&200 Value-Added Data Package

## 6.3 Configuring Basic OSPFv3 Functions

Before building OSPFv3 networks, you need to configure basic OSPFv3 functions.

### 6.3.1 Establishing the Configuration Task

You need to enable OSPFv3 and specify interfaces and area IDs before configuring other functions.

#### Applicable Environment

Enable the OSPFv3 process and specify its router ID before configuring OSPFv3; otherwise, other functions cannot take effect.

You must enable OSPFv3 and specify the interface and area ID before configuring other functions. OSPFv3 configurations, however, are independent of interface-related features.

#### Pre-configuration Tasks

Before configuring basic OSPFv3 functions, complete the following tasks:

- Making the network layers of the adjacent nodes accessible
- Enabling IPv6 capabilities

#### Data Preparation

To configure basic OSPFv3 functions, you need the following data.

| No. | Data                                                           |
|-----|----------------------------------------------------------------|
| 1   | Router ID                                                      |
| 2   | OSPFv3 process ID                                              |
| 3   | Interfaces on which OSPFv3 needs to be enabled and their areas |

### 6.3.2 Enabling OSPFv3

Creating an OSPFv3 process is a prerequisite for configuring all OSPFv3 features. By creating an OSPFv3 process, you can manually specify the router ID for a router.

#### Context

OSPFv3 supports multiple processes. Multiple OSPFv3 processes running on one router are differentiated by process IDs. OSPFv3 process ID is set when OSPFv3 is enabled and is only locally valid. It does not affect the packet exchange with other routers.



In the format of an IPv4 address, a router ID is a 32-bit unsigned integer that uniquely identifies a router within an AS. The router ID of OSPFv3 must be manually set. If no router ID is set, OSPFv3 fails to run normally.

When manually setting the router ID, ensure that the router IDs of any two routers in an AS are different. When multiple processes are enabled on a router, it is necessary to specify a unique route ID for each process.

To ensure the stable running of OSPFv3, you need to allocate router IDs and set them in network planning.

Do as follows on the router that runs OSPFv3.

## Procedure

**Step 1** Run:

```
system-view
```

The system view is displayed.

**Step 2** Run:

```
ospfv3 [process-id]
```

OSPFv3 is enabled and the OSPFv3 view is displayed.

**Step 3** Run:

```
router-id router-id
```

A Router ID is set.

----End

### 6.3.3 Enabling OSPFv3 on an Interface

For an interface with multiple instances, you need to specify which instance of the interface is enabled in the OSPFv3 process when enabling OSPFv3 on the interface.

## Context

After enabling OSPFv3 in the system view, you need to enable OSPFv3 on the interface.

Because an interface has multiple instances, you need to specify which instance of the interface is enabled in the OSPFv3 process when OSPFv3 is enabled on the interface. If no instance ID is specified, the value defaults to 0. The same instance must be enabled on the interfaces between which the neighbor relationship is set up.

Do as follows on the router that runs OSPFv3.

## Procedure

**Step 1** Run:

```
system-view
```

The system view is displayed.

**Step 2** Run:

```
interface interface-type interface-number
```

The interface view is displayed.

**Step 3** Run:

```
ospfv3 process-id area area-id [instance instance-id]
```

OSPFv3 is enabled on the interface.

The area ID can be a decimal integer or in the IPv4 address format, but it is displayed in the IPv4 address format.

**Step 4** (Optional) Run the **ospfv3 network-type { broadcast | nbma | p2mp [ non-broadcast ] | p2p }** [ instance instance-id ] command to configure the network type of an interface.

When an interface supports multi-instances, you must specify the value of *instance-id* when enabling OSPFv3 on the interface. If the value of *instance-id* is not specified, the default value 0 is adopted. In this case, the configured network type of an interface mismatches the actual network type of the interface. This step is mandatory in such a case.

---End

## 6.3.4 Entering the OSPFv3 Area View

By dividing an AS into different areas, specifying OSPFv3 interfaces, and specifying areas to which these interfaces belong, OSPFv3 can discover and calculate routes in an AS.

### Context

You must configure the routers in the same area based on the area. Otherwise, the neighbor routers cannot exchange information with each other. The congestion of routing information or routing loop is thus caused.

Do as follows on the router that runs OSPFv3.

### Procedure

**Step 1** Run:

```
system-view
```

The system view is displayed.

**Step 2** Run:

```
ospfv3 [process-id]
```

The OSPFv3 view is displayed.

**Step 3** Run:

```
area area-id
```

The OSPFv3 area view is displayed.

The area ID can be a decimal integer or in the IPv4 address format, but it is displayed in the IPv4 address format.

An OSPFv3 area cannot be deleted directly. Only after all the configurations in the area view are removed and the status of the related interfaces in this area become Down, this area is automatically removed.

---End

## 6.3.5 Checking the Configuration

After basic OSPFv3 functions are configured, you can check OSPFv3 brief information, LSDB information, neighbor information, and OSPFv3 routing table.

### Prerequisites

The configurations for the Basic OSPFv3 Functions are complete.

### Procedure

- Run the **display ospfv3** [ *process-id* ] command to check the summary information about the OSPFv3 process.
- Run the **display ospfv3** [ *process-id* ] **interface** [ **area** *area-id* ] [ *interface-type interface-number* ] command to check the OSPFv3 interface information.
- Run the commands as follow to check the LSDB information about OSPFv3:
  - **display ospfv3** [ *process-id* ] **lsdb** [ **area** *area-id* ] [ **originate-router** *advertising-router-id* | **self-originate** ] [ { **router** | **network** | **inter-router** [ **asbr-router** *asbr-router-id* ] | { **inter-prefix** | **nssa** } [ *ipv6-address prefix-length* ] | **link** | **intra-prefix** | **grace** } [ *link-state-id* ] ]
  - **display ospfv3** [ *process-id* ] **lsdb** [ **originate-router** *advertising-router-id* | **self-originate** ] **external** [ *ipv6-address prefix-length* ] [ *link-state-id* ]
- Run the **display ospfv3** [ *process-id* ] [ **area** *area-id* ] **peer** [ *interface-type interface-number* ] [ **verbose** ] command or **display ospfv3** [ *process-id* ] [ **area** *area-id* ] **peer neighbor-id** [ **verbose** ] command to check the information about the OSPFv3 neighbor.
- Run the commands as follow to check the OSPFv3 routing table:
  - **display ospfv3** [ *process-id* ] **routing**
  - **display ospfv3** [ *process-id* ] **routing** [ **abr-routes** | **asbr-routes** | **statistics** | *ipv6-address prefix-length* | **intra-routes** | **inter-routes** | **ase-routes** | **nssa-routes** ]
- Run the **display ospfv3** [ *process-id* ] **path** command to check the paths to a destination address.
- Run the **display default-parameter ospfv3** command to check the default OSPFv3 configuration.

----End

## 6.4 Establishing or Maintaining OSPFv3 Neighbor Relationship

By establishing and maintaining OSPFv3 neighbor relationships or adjacencies, you can build OSPFv3 networks.

### 6.4.1 Establishing the Configuration Task

When setting parameters on an interface, ensure that these parameters are consistent with those on the adjacent router.

## Applicable Environment

In applications, establishing or maintaining the OSPFv3 neighbor relationship is a premise for the construction of an OSPFv3 network. After the configuration in this section, you can:

- Adjust the convergence speed of the OSPFv3 network and network load posed by protocol packets by modifying OSPFv3 timers.
- Enable OSPFv3 to be disconnected from its neighbor when the number of OSPFv3 packet retransmissions exceeds the threshold by configuring Retransmission Limitation for OSPFv3. This prevents non-stop packet retransmissions if the neighbor does not receive packets.
- Speed up the convergence of an OSPFv3 network by adjusting the intervals for updating and receiving LSAs.

## Pre-configuration Tasks

Before establishing or maintaining the OSPFv3 neighbor relationship, complete the following tasks:

- Enabling IPv6 capability
- [Configuring Basic OSPFv3 Functions](#)

## Data Preparation

To establish or maintain the OSPFv3 neighbor relationship, you need the following data.

| No. | Data                                                 |
|-----|------------------------------------------------------|
| 1   | Interval for sending Hello packets                   |
| 2   | Dead time of the neighbor relationship               |
| 3   | Interval for retransmitting LSAs to adjacent routers |
| 4   | Delay in sending LSAs                                |

### 6.4.2 Configuring the Interval for Sending Hello Packets

By adjusting the Hello interval set on OSPFv3 neighbors, you can change the speed of establishing the neighbor relationship, thus changing the speed of network convergence.

#### Context

Hello packets are periodically sent to the neighbor router to detect and maintain the neighbor relationship and to elect the DR and the BDR. RFC 2328 requires that the Hello timer values of neighbors be consistent. The value of the Hello timer is inversely proportional to the route convergence speed and network load.

Do as follows on the router that runs OSPFv3.

## Procedure

- Step 1** Run:  
`system-view`  
The system view is displayed.
- Step 2** Run:  
`interface interface-type interface-number`  
The interface view is displayed.
- Step 3** Run:  
`ospfv3 timer hello interval [ instance instance-id ]`  
The interval for sending Hello packets is set on the interface.
- End

### 6.4.3 Configuring Dead Time of Neighbor Relationship

If a router does not receive a Hello packet from its neighbor within the Holddown time, the router considers the neighbor relationship invalid.

#### Context

If a router does not receive any Hello packet from its neighbor during a specified period, the neighbor router is considered invalid. The specified period is called the dead time of the neighbor relationship. The dead time must be at least four times the Hello interval on an interface.

Do as follows on the router that runs OSPFv3.

## Procedure

- Step 1** Run:  
`system-view`  
The system view is displayed.
- Step 2** Run:  
`interface interface-type interface-number`  
The interface view is displayed.
- Step 3** Run:  
`ospfv3 timer dead interval [ instance instance-id ]`  
The dead time of the neighbor relationship is specified.
- End

### 6.4.4 Configuring the Interval for Retransmitting LSAs to Neighboring

After a router sends an LSA to its neighbor, the router expects to receive an LSAck packet from its neighbor. If the router does not receive an LSAck packet within the LSA retransmission interval, it retransmits the LSA to the neighbor.

## Context

Do as follows on the router that runs OSPFv3.

## Procedure

### Step 1 Run:

```
system-view
```

The system view is displayed.

### Step 2 Run:

```
interface interface-type interface-number
```

The interface view is displayed.

### Step 3 Run:

```
ospfv3 timer retransmit interval [instance instance-id]
```

The interval for retransmitting LSAs to the adjacent routers is set.

The value of *seconds* must be greater than the time taken to transmit a packet between two routers.

#### NOTE

Do not set a value which is too small, for the interval between LSA retransmissions. Otherwise, unnecessary retransmissions may occur.

---End

## 6.4.5 Configuring the Delay for Transmitting LSAs on the Interface

It takes time to transmit OSPFv3 packets on a link. Therefore, a certain delay is added to the aging time of an LSA before the LSA is sent.

## Context

The LSA ages out in the LSDB of a local router instead of in the transmission process. You need to set the delay for an LSA before sending it. For a low-speed network, this configuration is necessary.

Do as follows on the router that runs OSPFv3.

## Procedure

### Step 1 Run:

```
system-view
```

The system view is displayed.

### Step 2 Run:

```
interface interface-type interface-number
```

The interface view is displayed.

### Step 3 Run:

```
ospfv3 trans-delay interval [instance instance-id]
```

The delay in transmitting LSAs on the interface is set.

---End

## 6.4.6 Checking the Configuration

After OSPFv3 neighbor relationships or adjacencies are stable, you can check OSPFv3 interface information and neighbor information.

### Prerequisites

The configurations for the Establishing or Maintaining OSPFv3 Neighbor Relationship are complete.

### Procedure

- Run the **display ospfv3** [*process-id*] **interface** [**area** *area-id*] [*interface-type interface-number*] command to check the OSPFv3 interface information.

---End

## 6.5 Configuring OSPFv3 Areas

OSPFv3 supports stub areas and virtual links, the principle and applicable environment of which are similar to those in OSPFv2.

### 6.5.1 Establishing the Configuration Task

Configuring a stub area is optional. Not all areas can be configured as stub areas. Generally, a stub area, which is located at the AS boundary, is a non-backbone area with only one ABR.

### Applicable Environment

To reduce the number of LSAs in the network and enhance OSPFv3 extensibility, define OSPFv3 areas. For some non-backbone areas at the edge of ASs, you can define them as stub areas for further reducing the size of the routing table and the number of LSAs.

### Pre-configuration Tasks

Before configuring OSPFv3 area attributes, complete the following tasks:

- Enabling IPv6 capability
- [Configuring Basic OSPFv3 Functions](#)

### Data Preparation

To configure OSPFv3 area attributes, you need the following data.

| No. | Data                                         |
|-----|----------------------------------------------|
| 1   | Areas to be defined as stub areas            |
| 2   | Metrics of default routes sent to stub areas |

## 6.5.2 Configuring OSPFv3 Stub Areas

A stub area is a special area in which ABRs do not flood the received AS external routes. Thus, the number of LSAs is greatly reduced.

### Context

Do as follows on each router that runs OSPFv3 in the stub area:

### Procedure

**Step 1** Run:

```
system-view
```

The system view is displayed.

**Step 2** Run:

```
ospfv3 [process-id]
```

The OSPFv3 view is displayed.

**Step 3** Run:

```
area area-id
```

The OSPFv3 area view is displayed.

**Step 4** Run:

```
stub [no-summary]
```

The area is configured as a stub area.

**Step 5** (Optional) Run:

```
default-cost cost
```

The cost of the default route sent to the stub area is set.

By default, the cost of the default route sent to the stub area is 1.

This command is configured on the ABR of the stub area only to set the cost of the default route to be sent to the stub area. This command does not need to be configured on other routers in the stub area.

The parameter **no-summary** takes effect only when the **stub** command is configured on the ABR. If this parameter is configured, the ABR only sends the summary-LSA of a default route to the stub area without originating other summary-LSAs. The stub area without AS-external-LSAs or Summary-LSAs is called a totally stub area.

----End

## 6.5.3 Configuring OSPFv3 Virtual Links

You can establish the logical connectivity between backbone areas and the non-backbone areas that are not physically connected to the backbone area.

### Context

After OSPFv3 areas are defined, OSPFv3 route update between non-backbone areas is implemented through a backbone area. Then, OSPFv3 requires that all non-backbone areas



should maintain the connectivity with the backbone area and the backbone area should maintain its own connectivity. In actual applications, this requirement may not be met because of some restrictions. To solve this problem, you can configure OSPFv3 virtual links.

A virtual link must be configured at both ends of the link; otherwise, it does not take effect.

Do as follows on the router that runs OSPFv3.

## Procedure

### Step 1 Run:

```
system-view
```

The system view is displayed.

### Step 2 Run:

```
ospfv3 [process-id]
```

The OSPFv3 view is displayed.

### Step 3 Run:

```
area area-id
```

The OSPFv3 area view is displayed.

### Step 4 Run:

```
vlink-peer router-id [hello hello-interval | retransmit retransmit-interval |
trans-delay trans-delay-interval | dead dead-interval | instance instance-id] *
```

A virtual link is created and configured.

---End

## 6.5.4 Checking the Configuration

After OSPFv3 area attributes are configured, you can check the OSPFv3 LSDB, routing table, and virtual links.

## Prerequisites

The configurations for the OSPFv3 Areas are complete.

## Procedure

- Run the commands as follow to check the LSDB information about OSPFv3:
  - **display ospfv3** [ process-id ] **lsdb** [ area area-id ] [ originate-router advertising-router-id | self-originate ] [ { router | network | inter-router [ asbr-router asbr-router-id ] | { inter-prefix | nssa } [ ipv6-address prefix-length ] | link | intra-prefix | grace } [ link-state-id ] ]
  - **display ospfv3** [ process-id ] **lsdb** [ originate-router advertising-router-id | self-originate ] **external** [ ipv6-address prefix-length ] [ link-state-id ]
- Run the commands as follow to check the OSPFv3 routing table:
  - **display ospfv3** [ process-id ] **routing**

- **display ospfv3** [ *process-id* ] **routing** [ **abr-routes** | **asbr-routes** | **statistics** [ **uninstalled** ] | *ipv6-address prefix-length* | **intra-routes** | **inter-routes** | **ase-routes** | **nssa-routes** ]
- Run the **display ospfv3** [ *process-id* ] **vlink** command to check the information about OSPFv3 virtual links.

----End

## 6.6 Configuring OSPFv3 NSSA Areas

By configuring areas as NSSA areas, external routes can be imported, and a new type of LSA, namely, Type 7 NSSA LSA is introduced.

### 6.6.1 Establishing the Configuration Task

NSSAs are introduced because stub areas cannot import external routes. An NSSA allows the transmission of Type 7 LSAs.

#### Applicable Environment

An NSSA allows the transmission of Type 7 LSAs, which are generated by ASBRs in an NSSA. The Type 7 LSAs converting into Type 5 LSAs in the NSSA and advertised to other areas.

#### Pre-configuration Tasks

Before configuring an OSPFv3 NSSA, complete the following tasks:

- Configuring IP addresses for interfaces to make neighboring nodes reachable
- [Configuring basic OSPFv3 functions](#)

#### Data Preparation

To configure an OSPFv3 NSSA, you need the following data.

| No. | Data                                      |
|-----|-------------------------------------------|
| 1   | Cost of the default route sent to an NSSA |

### 6.6.2 Defining the Current Area to Be an NSSA Area

Derived from a stub area, an NSSA allows AS external routes to be imported; an ASBR advertises Type 7 NSSA LSAs in the local NSSA.

#### Context

Do as follows on the OSPFv3 router:

## Procedure

**Step 1** Run:

```
system-view
```

The system view is displayed.

**Step 2** Run:

```
ospfv3 [process-id]
```

The OSPFv3 process view is displayed.

**Step 3** Run:

```
area area-id
```

The OSPFv3 area view is displayed.

**Step 4** Run:

```
nssa [default-route-advertise [cost cost | type type | tag tag] * | no-import-
route | no-summary | translator-always | translator-interval translator-interval |
set-n-bit] *
```

An area is configured as an NSSA.

----End

## Follow-up Procedure

To connect routers to an NSSA, you need to run the **nssa** command to configure NSSA attributes for the area to which the routers belong.

The area may be updated after NSSA attributes are configured or deleted. Thus, the NSSA attributes can be re-configured or deleted only after the last update of NSSA attributes is complete.

## 6.6.3 Checking the Configuration

After OSPFv3 NSSAs are configured, you can check OSPFv3 routing table information.

## Prerequisites

The configurations for OSPFv3 NSSAs are complete.

## Procedure

- Run the **display ospfv3 area** command to check information about OSPFv3 areas.
- Run the commands as follow to check the OSPFv3 routing table.
  - **display ospfv3 [ process-id ] routing**
  - **display ospfv3 [ process-id ] routing [ abr-routes | asbr-routes | statistics | ipv6-address prefix-length | intra-routes | inter-routes | ase-routes | nssa-routes ]**

----End

## 6.7 Configuring OSPFv3 Route Attributes

By setting OSPFv3 route attributes, you can change OSPFv3 routing policies to meet the requirements of complex networks.

### 6.7.1 Establishing the Configuration Task

Before configuring OSPFv3 route attributes, familiarize yourself with the applicable environment, complete pre-configuration tasks, and obtain the required data. This can help you complete the configuration task quickly and accurately.

#### Applicable Environment

In actual applications, to meet the requirements of a complicated networking environment, you can change OSPFv3 routing policies by configuring OSPFv3 route attributes. Through the following procedures, you can:

- Set the cost on the OSPFv3 interface.
- Configure load balancing among equal-cost routes.

#### Pre-configuration Tasks

Before configuring OSPFv3 route attributes, complete the following tasks:

- Enabling IPv6 capability
- [Configuring Basic OSPFv3 Functions](#)

#### Data Preparation

To configure OSPFv3 route attributes, you need the following data.

| No. | Data                                |
|-----|-------------------------------------|
| 1   | Link cost                           |
| 2   | Maximum number of equal-cost routes |

### 6.7.2 Setting the Cost of the OSPFv3 Interface

OSPFv3 can automatically calculate the link cost for an interface according to the interface bandwidth. You can also set the link cost for the interface by using the related command.

#### Context

You can control route calculation by setting the link cost of OSPFv3 on different interfaces.

Do as follows on the router that runs OSPFv3.

## Procedure

- Step 1** Run:  
`system-view`  
The system view is displayed.
- Step 2** Run:  
`interface interface-type interface-number`  
The interface view is displayed.
- Step 3** Run:  
`ospfv3 cost cost [ instance instance-id ]`  
The cost is set on the OSPFv3 interface.  
By default, the link cost on an OSPFv3 interface is 1.  
----End

### 6.7.3 Setting the Maximum Number of OSPFv3 Equal-Cost Routes

If the destinations and costs of the multiple routes discovered by one routing protocol are the same, load balancing can be performed among these routes.

## Context

Do as follows on the router that runs OSPFv3:

## Procedure

- Step 1** Run:  
`system-view`  
The system view is displayed.
- Step 2** Run:  
`ospfv3 [ process-id ]`  
The OSPFv3 view is displayed.
- Step 3** Run:  
`maximum load-balancing number`  
The maximum number of equal-cost routes is set.  
The value is an integer ranging from 1 to 32. The default value is 32.  
----End

### 6.7.4 Checking the Configuration

After OSPFv3 route attributes are configured, you can check the OSPFv3 interface, LSDB, and routing table.

## Prerequisites

The configurations for the OSPFv3 Route Attributes are complete.

## Procedure

- Run the **display ospfv3**[ *process-id* ] **interface** [ **area** *area-id* ] [ *interface-type* *interface-number* ] command to check the OSPFv3 interface information.
- Run the commands as follow to check the LSDB information about OSPFv3:
  - **display ospfv3** [ *process-id* ] **lsdb** [ **area** *area-id* ] [ **originate-router** *advertising-router-id* | **self-originate** ] [ { **router** | **network** | **inter-router** [ **asbr-router** *asbr-router-id* ] | { **inter-prefix** | **nssa** } [ *ipv6-address prefix-length* ] | **link** | **intra-prefix** | **grace** } [ *link-state-id* ] ]
  - **display ospfv3** [ *process-id* ] **lsdb** [ **originate-router** *advertising-router-id* | **self-originate** ] **external** [ *ipv6-address prefix-length* ] [ *link-state-id* ]
- Run the commands as follow to check the OSPFv3 routing table:
  - **display ospfv3** [ *process-id* ] **routing**
  - **display ospfv3** [ *process-id* ] **routing** [ **abr-routes** | **asbr-routes** | **statistics** | *ipv6-address prefix-length* | **intra-routes** | **inter-routes** | **ase-routes** | **nssa-routes** ]

----End

## 6.8 Controlling OSPFv3 Routing Information

This section describes how to control OSPFv3 routing information. Detailed operations include configuring route aggregation, filtering the received routes, and importing external routes.

### 6.8.1 Establishing the Configuration Task

Before controlling OSPFv3 routing information, familiarize yourself with the applicable environment, complete pre-configuration tasks, and obtain the required data. This can help you complete the configuration task quickly and accurately.

#### Applicable Environment

Through the configuration in this section, you can control the advertising and receiving of OSPFv3 routing information and configure OSPFv3 to import external routes.

#### Pre-configuration Tasks

Before controlling OSPFv3 routing information, complete the following tasks:

- Enabling IPv6 capability
- [Configuring Basic OSPFv3 Functions](#)

#### Data Preparation

To control OSPFv3 routing information, you need the following data.

| No. | Data                                                      |
|-----|-----------------------------------------------------------|
| 1   | Prefix of IPv6 routes after aggregation                   |
| 2   | Filtering list or name used to filter routing information |

| No. | Data                                                           |
|-----|----------------------------------------------------------------|
| 3   | Link cost on an OSPFv3 interface                               |
| 4   | Maximum number of equal-cost routes                            |
| 5   | Name, process ID, and metric of external routes to be imported |

## 6.8.2 Configuring OSPFv3 Route Aggregation

An ABR can summarize routes with the same prefix into one LSA and advertise the summarized route in other areas. An ASBR can also summarize imported routes with the same prefix into one LSA and then advertise the summarized route to other areas. This can reduce the size of the LSDB in other areas.

### Context

If multiple continuous network segments exist in this area, use the **abr-summary** command to summarize them into one network segment. In this way, the ABR only sends an LSA after summarization. No LSA that belongs to the summarization network segment is separately transmitted, thus reducing the LSDB size of other areas.

When a large number of routes are imported, use the **asbr-summary** command to summarize the imported routes and set the delay for advertising the summarized route. In this manner, the summarized route advertised each time contains more valid routing information, and network flapping caused by incorrect routing information is avoided.

### Procedure

- Configure route summarization on an ABR.

Do as follows on the ABR that runs OSPFv3:

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
ospfv3 [process-id]
```

The OSPFv3 view is displayed.

3. Run:

```
area area-id
```

The OSPFv3 area view is displayed.

4. Run:

```
abr-summary ipv6-address prefix-length [cost cost | not-advertise]*
```

Route summarization is configured in the OSPFv3 area.

**cost** *cost* set the cost of a summarized route. By default, the cost of a summarized route is the maximum cost among those of routes that are summarized. The value ranges from 1 to 16777214.

If **not-advertise** is set, no routing information of the network segment is advertised.

- Configure route summarization on an ASBR.

Do as follows on the ASBR that runs OSPFv3:

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
ospfv3 [process-id]
```

The OSPFv3 view is displayed.

3. Run:

```
asbr-summary ipv6-address prefix-length [cost cost | tag tag | not-
advertise | distribute-delay interval] *
```

Route summarization is configured on the ASBR.

**cost** *cost* specifies the cost of a summarized route. By default, the cost of a summarized route is the maximum cost among those of routes that are summarized. The value ranges from 1 to 16777214.

**tag** *tag* specifies the tag used to control route advertisement. The value of this parameter ranges from 1 to 4294967295.

If **not-advertise** is specified in the command, the summarized IPv6 route that matches a specified IPv6 prefix or prefix length is not advertised.

**distribute-delay** *interval* specifies the delay for advertising a summarized route.

---End

## 6.8.3 Configuring OSPFv3 to Filter the Received Routes

By configuring filtering conditions for routing information, you can allow only the routes that pass the filtering to be received or advertised.

### Context

After receiving LSAs, OSPFv3 determines whether to add the calculated routes to the local routing table according to the filtering policy.

Do as follows on the router that runs OSPFv3.

### Procedure

- Step 1** Run:

```
system-view
```

The system view is displayed.

- Step 2** Run:

```
ospfv3 [process-id]
```

The OSPFv3 view is displayed.



**Step 3** Run:

```
filter-policy ipv6-prefix ipv6-prefix-name import
```

OSPFv3 is configured to filter the imported routes.

Using the **filter-policy** command, you can only filter the routes calculated by OSPFv3. Routes that do not pass the filtering are neither added to the OSPFv3 routing table nor advertised.

----End

## 6.8.4 Configuring OSPFv3 to Import External Routes

Importing the routes discovered by other routing protocols can enrich OSPFv3 routing information.

### Context

Because OSPFv3 is a link state-based routing protocol and cannot directly filter the advertised LSAs, OSPFv3 must filter the routes when importing them. Then, only the routes that pass the filtering can be advertised.

Do as follows on the router that runs OSPFv3.

### Procedure

**Step 1** Run:

```
system-view
```

The system view is displayed.

**Step 2** Run:

```
ospfv3 [process-id]
```

The OSPFv3 view is displayed.

**Step 3** Run:

```
default { cost cost | tag tag | type type } *
```

The default cost of the imported route is set.

**Step 4** Run:

```
import-route protocol [process-id] [cost cost | type type | tag tag | route-policy route-policy-name] *
```

External routes are imported.

**Step 5** (Optional) Run:

```
default-route-advertise [always | cost cost | type type | tag tag | route-policy route-policy-name] *
```

Default routes are advertised to the OSPFv3 route area.

**Step 6** (Optional) Run:

```
filter-policy ipv6-prefix ipv6-prefix-name export [protocol [process-id]]
```

The imported external routes are filtered.

After you run the **import-route** command on an OSPFv3 router to import external routes, the router becomes an ASBR.

You can configure OSPFv3 to filter a certain type of routing information by specifying the *protocol*. If *protocol* is not specified, OSPFv3 filters all the imported routes.

 **NOTE**

The **filter-policy** command takes effect only on the routes imported through the **import-route** command by the ASBR, that is, filters the imported routes. The routes that are filtered out do not generate LSAs and cannot be advertised by OSPFv3. If the **import-route** command is not configured to import other external routes (including OSPFv3 routes in different processes), the **filter-policy** command does not take effect.

---End

## 6.8.5 Checking the Configuration

After OSPFv3 route attributes are configured, you can check the OSPFv3 interface, LSDB, and routing table.

### Prerequisites

The configurations for Controlling OSPFv3 Routing Information are complete.

### Procedure

- Run the commands as follow to check the OSPFv3 route aggregation:
  - **display ospfv3** [ *process-id* ] **abr-summary-list** [ *ipv6-address prefix-length* ]
  - **display ospfv3** [ *process-id* ] **asbr-summary** [ *ipv6-address prefix-length* ] [ *verbose* ]
- Run the commands as follow to check the LSDB information about OSPFv3:
  - **display ospfv3** [ *process-id* ] **lsdb** [ *area area-id* ] [ **originate-router** *advertising-router-id* | **self-originate** ] [ { **router** | **network** | **inter-router** [ **asbr-router** *asbr-router-id* ] | { **inter-prefix** | **nssa** } [ *ipv6-address prefix-length* ] | **link** | **intra-prefix** | **grace** } [ *link-state-id* ] ]
  - **display ospfv3** [ *process-id* ] **lsdb** [ **originate-router** *advertising-router-id* | **self-originate** ] **external** [ *ipv6-address prefix-length* ] [ *link-state-id* ]
- Run the commands as follow to check the OSPFv3 routing table:
  - **display ospfv3** [ *process-id* ] **routing**
  - **display ospfv3** [ *process-id* ] **routing** [ **abr-routes** | **asbr-routes** | **statistics** | *ipv6-address prefix-length* | **intra-routes** | **inter-routes** | **ase-routes** | **nssa-routes** ]

----End

## 6.9 Optimizing an OSPFv3 Network

By configuring OSPFv3 functions in special network environments, you can adjust and optimize the OSPFv3 network performance.

### 6.9.1 Establishing the Configuration Task

Before optimizing an OSPFv3 network, familiarize yourself with the applicable environment, complete pre-configuration tasks, and obtain the required data. This can help you complete the configuration task quickly and accurately.

## Applicable Environment

By adjusting the OSPFv3 timer, you can change the convergence speed of an OSPFv3 network and the network overload caused by protocol packets. On low-speed links, you need to consider the delay in transmitting LSAs on the interface. By adjusting the SPF calculation interval, you can mitigate resource consumption due to frequent network changes.

You can specify the DR priority of an interface to affect the DR/BDR election in a broadcast network.

## Pre-configuration Tasks

Before optimizing an OSPFv3 network, complete the configuration tasks:

- Enabling IPv6 capability
- [Configuring Basic OSPFv3 Functions](#)

## Data Preparation

To optimize an OSPFv3 network, you need the following data.

| No. | Data                         |
|-----|------------------------------|
| 1   | Values of OSPFv3 timers      |
| 2   | Values of SPF timers         |
| 3   | DR priority of the interface |

## 6.9.2 Configuring the SPF Timer

By setting the interval for SPF calculation, you can reduce resource consumption caused by frequent network changes.

### Context

Whenever the LSDB of OSPFv3 changes, the shortest path should be recalculated. Calculating the shortest path each time the LSDB changes consumes enormous resources and lowers the efficiency of a router.

Adjusting the SPF delay and hold interval can suppress frequent network changes to avoid resource consumption.

Do as follows on the router that runs OSPFv3.

### Procedure

- Configure an SPF normal timer.

Do as follows on the router that runs OSPFv3:

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
ospfv3 [process-id]
```

The OSPFv3 view is displayed.

3. Run:

```
spf timers delay-interval hold-interval
```

An SPF normal timer is configured.

- Configure an SPF intelligent timer.

Do as follows on the router that runs OSPFv3:

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
ospfv3 [process-id]
```

The OSPFv3 view is displayed.

3. Run:

```
spf-schedule-interval { delay-interval hold-interval | intelligent-timer
max-interval start-interval hold-interval-1 }
```

An SPF intelligent timer is configured.

 **NOTE**

An SPF normal timer and an SPF intelligent timer are mutually exclusive.

---End

## 6.9.3 Setting the Interval for Receiving LSAs

Setting the interval for receiving LSAs prevents unnecessary LSA updates.

### Context

When a network is instable, control the minimum interval for receiving the same LSA update. To prevent unnecessary LSA updates caused by network changes, by default, set the interval for receiving the same LSA update to 1000ms.

Do as follows on the router that runs OSPFv3.

### Procedure

- Step 1** Run:

```
system-view
```

The system view is displayed.

- Step 2** Run:

```
ospfv3 [process-id]
```

The OSPFv3 view is displayed.

**Step 3** Run:

```
lsa-arrival-interval arrival-interval
```

The interval for receiving LSAs is set.

*arrival-interval* is an integer ranging from 1 to 10000, in milliseconds. By default, the interval for receiving LSAs is 1000ms.

----End

## 6.9.4 Configuring an Intelligent Timer for Generating LSAs

Configuring an intelligent timer for generating LSAs speeds up network convergence.

### Context

Setting the millisecond-level interval for generating the same LSA speeds up network convergence. When a network becomes instable, reduce the interval for generating the same LSA by using an intelligent timer.

Do as follows on the router that runs OSPFv3.

### Procedure

**Step 1** Run:

```
system-view
```

The system view is displayed.

**Step 2** Run:

```
ospfv3 [process-id]
```

The OSPFv3 view is displayed.

**Step 3** Run:

```
lsa-originate-interval intelligent-timer max-interval start-interval hold-interval
```

The interval for generating the same LSA is set.

- *max-interval* specifies the maximum interval for updating LSAs. The value ranges from 1 to 10000, in milliseconds.
- *start-interval* specifies the initial interval for updating LSAs. The value ranges from 0 to 1000, in milliseconds.
- *hold-interval* specifies the hold interval for updating LSAs. The value ranges from 1 to 5000, in milliseconds.

By default, the maximum interval for updating LSAs is 5000ms, the initial interval for updating LSAs is 500ms, the hold interval for updating LSAs is 1000ms.

----End

## 6.9.5 Suppressing an Interface from Sending and Receiving OSPFv3 Packets

By suppressing the OSPFv3 interface from receiving and sending OSPFv3 packets, you can prevent routers on a certain network from obtaining OSPFv3 routing information and prevent the local router from receiving routing information from other routers.

## Context

To prevent a router from advertising routes to the router on a certain network and from importing the routes of other routers, you can suppress the interface on which OSPFv3 is enabled from receiving and sending OSPFv3 packets.

Do as follows on the router that runs OSPFv3.

## Procedure

### Step 1 Run:

```
system-view
```

The system view is displayed.

### Step 2 Run:

```
ospfv3 [process-id]
```

The OSPFv3 view is displayed.

### Step 3 Run:

```
silent-interface interface-type interface-number
```

The interface is suppressed from sending and receiving OSPFv3 packets.

----End

## Follow-up Procedure

Different processes can suppress the same interface from sending and receiving OSPFv3 packets, but the **silent-interface** command is valid only for the OSPFv3 interface on which the specified process is enabled, and does not take effect on the interface of other processes.

After an OSPFv3 interface is set to be silent, the interface can still advertise its direct routes through the Intra-Area-Prefix-LSA of the same router. No OSPFv3 neighbor relationship can be set up on the interface. Therefore, the OSPFv3 adaptability is enhanced.

## 6.9.6 Configuring DR Priority of an Interface

When configuring a broadcast network or an NBMA network, you can specify the DR priority for each interface to change the results of DR/BDR election on the network.

## Context

The DR priority on a router interface qualifies the interface for the DR election. If the DR priority is 0, the router cannot be elected as a DR or BDR.

Do as follows on the router that runs OSPFv3.

## Procedure

### Step 1 Run:

```
system-view
```

The system view is displayed.

**Step 2** Run:

```
interface interface-type interface-number
```

The interface view is displayed.

**Step 3** Run:

```
ospfv3 dr-priority priority [instance instance-id]
```

The DR priority of the interface is set.

----End

## Follow-up Procedure

After the DR priority is changed, you can re-elect a DR or BDR through the following methods, which, however, will result in the interruption of the OSPFv3 neighbor relationship between routers and therefore are used only when necessary.

- Restarting all routers.
- Running the **shutdown** and **undo shutdown** commands on the interface on which the OSPFv3 neighbor relationship is set up.

## 6.9.7 Configuring Stub Routers

When a router has a heavy load and cannot forward any other packets, you can configure it as a stub router. After the router is configured as a stub router, other OSPFv3 routers do not use this router to forward data but they can have a route to this stub router.

### Context

A stub router is used to control traffic. It notifies OSPFv3 routers not to forward data by the stub router, but they can have a route to the stub router.

Do as follows on the router that runs OSPFv3:

### Procedure

**Step 1** Run:

```
system-view
```

The system view is displayed.

**Step 2** Run:

```
ospfv3 [process-id]
```

The OSPFv3 process view is displayed.

**Step 3** Run:

```
stub-router [on-startup [interval]]
```

The stub router is configured.

 **NOTE**

There is no correlation between the stub router configured through this command and the router in the stub area.

----End

## 6.9.8 Ignoring MTU Check on DD Packets

By disabling an interface from checking the MTU field in the received DD packet, you can enable an OSPFv3 router to receive the packet with the MTU field being 0.

### Context

Do as follows on the router that runs OSPFv3:

### Procedure

**Step 1** Run:

```
system-view
```

The system view is displayed.

**Step 2** Run:

```
interface interface-type interface-number
```

The interface view is displayed.

**Step 3** Run:

```
ospfv3 mtu-ignore [instance instance-id]
```

The MTU check on DD packets is ignored.

After the command is used, the interface does not check the MTU field of a received DD packet.

----End

## 6.9.9 Checking the Configuration

After an OSPFv3 network is optimized, you can check the OSPFv3 interface, LSDB, and routing table.

### Prerequisites

The configurations for Optimizing an OSPFv3 Network are complete.

### Procedure

- Run the **display ospfv3** [*process-id*] **interface** [ **area** *area-id* ] [ *interface-type interface-number* ] command to check the OSPFv3 interface information.
- Run the commands as follow to check the LSDB information about OSPFv3:
  - **display ospfv3** [*process-id*] **lsdb** [ **area** *area-id* ] [ **originate-router** *advertising-router-id* | **self-originate** ] [ { **router** | **network** | **inter-router** [ **asbr-router** *asbr-router-id* ] | { **inter-prefix** | **nssa** } [ *ipv6-address prefix-length* ] | **link** | **intra-prefix** | **grace** } [ *link-state-id* ] ]
  - **display ospfv3** [*process-id*] **lsdb** [ **originate-router** *advertising-router-id* | **self-originate** ] **external** [ *ipv6-address prefix-length* ] [ *link-state-id* ]
- Run the commands as follow to check the OSPFv3 routing table:
  - **display ospfv3** [*process-id*] **routing**



- **display ospfv3** [ *process-id* ] **routing** [ **abr-routes** | **asbr-routes** | **statistics** | *ipv6-address prefix-length* | **intra-routes** | **inter-routes** | **ase-routes** | **nssa-routes** ]

----End

## 6.10 Configuration OSPFv3 GR

By configuring OSPFv3 GR, you can avoid inaccurate route calculation and packet loss after an OSPFv3 router restarts.

### 6.10.1 Establishing the Configuration Task

By default, the OSPFv3 GR capability and Helper capability are disabled.

#### Applicable Environment

To prevent route flapping and service interruption due to the restart of OSPFv3, you can enable OSPFv3 GR.

After OSPFv3 restarts, the GR restarter and the GR helper keep the neighbor relationship, exchange routing information, synchronize the database, and update the routing table and the forwarding table. OSPFv3 fast convergence is thus realized.

#### NOTE

The AR150/200 can function as only the Helper router, but cannot function as the Restarter router.

#### Pre-configuration Tasks

Before configuring OSPFv3 GR, complete the following task:

- [Configuring Basic OSPFv3 Functions](#)

#### Data Preparation

To configure OSPFv3 GR, you need the following data.

| No. | Data                                              |
|-----|---------------------------------------------------|
| 1   | OSPFv3 process ID                                 |
| 2   | Filtering rule of the helper mode of OSPFv3 peers |

### 6.10.2 Enabling OSPFv3 GR

After an OSPFv3 process restarts through GR, the Restarter and the Helper reestablish the neighbor relationship, exchange routing information, synchronize the LSDB, and update the routing table and forwarding table. These operations help ensure OSPFv3 fast convergence and stabilize the network topology.

#### Context

Do as follows on the router that runs OSPFv3:

## Procedure

**Step 1** Run:

```
system-view
```

The system view is displayed.

**Step 2** Run:

```
ospfv3 process-id
```

The OSPFv3 view is displayed.

**Step 3** Run:

```
graceful-restart [period period | ack-time time | retransmit-interval interval |
lsa-checking-ignore | planned-only] *
```

OSPFv3 GR is enabled.

By default, OSPFv3 GR is disabled.

**ack-time** is optional. After **ack-time** is specified, the restarter can discover more neighbors in the *time* period.

----End

## 6.10.3 Enabling the Helper of OSPFv3 GR

The GR Helper, which is a neighbor of the GR Restarter, can identify GR signaling, maintain the adjacency with the Restarter during the active/standby switchover of the Restarter, and help the Restarter to restore the network topology.

### Context

Do as follows on the router that runs OSPFv3:

## Procedure

**Step 1** Run:

```
system-view
```

The system view is displayed.

**Step 2** Run:

```
ospfv3 process-id
```

The OSPFv3 view is displayed.

**Step 3** Run:

```
helper-role [{ ip-prefix ip-prefix-name | acl-number acl-number | acl-name acl-
name } | max-grace-period period | planned-only | lsa-checking-ignore] *
```

The helper of OSPFv3 GR is enabled.

By default, the helper of OSPFv3 GR is disabled.

----End

## 6.10.4 Check the Configuration

After OSPFv3 GR is configured, you can check GR information.

### Prerequisites

The configurations for OSPFv3 GR are complete.

### Procedure

- Run the **display ospfv3 [ *process-id* ] graceful-restart-information** command to check the status of OSPFv3 GR.

---End

## 6.11 Configuring the Network Management Function of OSPFv3

OSPFv3 supports the network management function. You can bind the OSPFv3 MIB to a certain OSPFv3 process.

### 6.11.1 Establishing the Configuration Task

Before configuring the network management function for OSPFv3, familiarize yourself with the applicable environment, complete pre-configuration tasks, and obtain the required data. This can help you complete the configuration task quickly and accurately.

#### Applicable Environment

OSPFv3 supports the network management function. You can bind OSPFv3 MIB and a certain OSPFv3 process. In addition, OSPFv3 also supports the trap function and the log function.

#### Pre-configuration Tasks

Before configuring the network management function of OSPFv3, complete the following tasks:

- Configuring IP addresses for interfaces to make neighboring nodes reachable
- [Configuring Basic OSPFv3 Functions](#)

#### Data Preparation

None.

### 6.11.2 Configuring OSPFv3 MIB Binding

The MIB is a virtual database of the device status maintained by the managed devices.

#### Context

When multiple OSPFv3 processes are enabled, you can configure OSPFv3 MIB to select the process to be processed, that is, that is, configure OSPFv3 MIB to select the process to which it is bound.

Do as follows on the OSPFv3 router.

## Procedure

**Step 1** Run:

```
system-view
```

The system view is displayed.

**Step 2** Run:

```
ospfv3 mib-binding process-id
```

OSPFv3 MIB binding is configured.

----End

## 6.11.3 Configuring OSPFv3 Trap

Traps are the notifications sent from a router to inform the NMS of the fault detected by the system.

### Context

Do as follows on the OSPFv3 router.

## Procedure

**Step 1** Run:

```
system-view
```

The system view is displayed.

**Step 2** Run:

```
snmp-agent trap enable feature-name ospfv3 [trap-name { ifconfigerror |
ifrxbadpacket | ifstatechange | nbrrestarthelperstatuschange | nbrstatechange |
nssatranslatorstatuschange | restartstatuschange | virtifconfigerror |
virtifrxbadpacket | virtifstatechange | virtnbrrestarthelperstatuschange |
virtnbrstatechange }]
```

The trap function for the OSPFv3 module is enabled.

----End

## 6.11.4 Check the Configuration

After the network management function is configured for OSPFv3, you can check the contents of the information channel, and information recorded in the information center, log buffer, and trap buffer.

### Prerequisites

The configurations of the Network Management Function of OSPFv3 are complete.

## Procedure

- Run the **display current-configuration** command to check the configuration parameters currently validated on the router.

----End

## 6.12 Maintaining OSPFv3

Maintaining OSPFv3 and Debugging OSPFv3 involve resetting OSPFv3.

### 6.12.1 Resetting OSPFv3

Restarting OSPFv3 can reset OSPFv3. In addition, you can reset OSPFv3 through GR.

#### Context



#### CAUTION

The OSPFv3 adjacency is removed when you reset the OSPFv3 connection by using the **reset ospfv3** command. Exercise caution when running this command.

---

After modifying the OSPFv3 routing policy or protocol, reset the OSPFv3 connection to validate the modification. To reset OSPFv3 connections, run the following **reset ospfv3** command in the user view.

#### Procedure

- To validate the new configuration, run the following commands:
  - **reset ospfv3** { *process-id* | **all** } [ **graceful-restart** [ **extend-period** *period* ] ]
  - **reset ospfv3** { *process-id* | **all** } **counters** [ **neighbor** [ *interface-type interface-number* ] [ *router-id* ] ]

----End

## 6.13 Configuration Examples

This section provides several configuration examples of OSPFv3 together with the configuration flowchart. The configuration examples explain networking requirements, configuration notes, and configuration roadmap.

### 6.13.1 Example for Configuring OSPFv3 Areas

This part provides an example for configuring basic OSPFv3 functions. Detailed operations include enabling OSPFv3 on each router and specifying network segments in different areas.

## Networking Requirements

As shown in **Figure 6-1**, all routers run OSPFv3. The entire autonomous system is divided into three areas. Router B and Router C serve as ABRs to forward the inter-area routes.

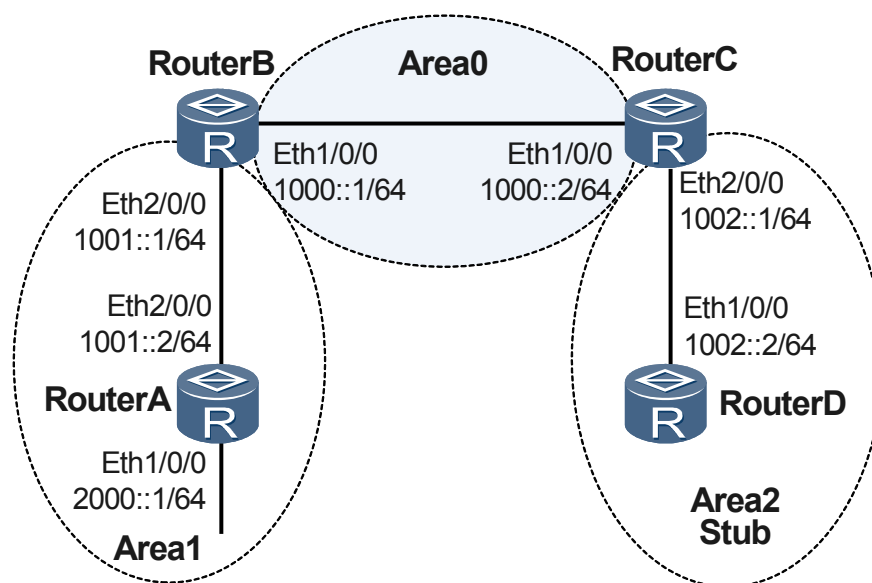
It is required that Area 2 be configured as a stub area to decrease the LSAs advertised to this area, without affecting route reachability.



**NOTE**

AR150/200 is RouterD.

**Figure 6-1** Networking diagram of configuring OSPFv3 areas



## Configuration Roadmap

The configuration roadmap is as follows:

1. Enable basic OSPFv3 function on each router.
2. Configure Area 2 as a stub area and check the OSPFv3 routing table of Router D.
3. Configure Area 2 as a totally stub area and check the OSPFv3 routing table of Router D.

## Data Preparation

To complete the configuration, you need the following data:

- Router ID of Router A as 1.1.1.1 of Area 1
- Router ID of Router B as 2.2.2.2 of Areas 0 and 1
- Router ID of Router C as 3.3.3.3 of Areas 0 and 2
- Router ID of Router D as 4.4.4.4 of Area 2

## Procedure

**Step 1** Assign an IPv6 address for each interface.

The details are not mentioned here.

**Step 2** Configure basic OSPFv3 functions.

# Configure Router A.

```
[RouterA] ipv6
[RouterA] ospfv3
[RouterA-ospfv3-1] router-id 1.1.1.1
[RouterA-ospfv3-1] quit
[RouterA] interface ethernet 1/0/0
[RouterA-Ethernet1/0/0] ospfv3 1 area 1
[RouterA-Ethernet1/0/0] quit
[RouterA] interface ethernet 2/0/0
[RouterA-Ethernet2/0/0] ospfv3 1 area 1
[RouterA-Ethernet2/0/0] quit
```

# Configure Router B.

```
[RouterB] ipv6
[RouterB] ospfv3
[RouterB-ospfv3-1] router-id 2.2.2.2
[RouterB-ospfv3-1] quit
[RouterB] interface ethernet 1/0/0
[RouterB-Ethernet1/0/0] ospfv3 1 area 0
[RouterB-Ethernet1/0/0] quit
[RouterB] interface ethernet 2/0/0
[RouterB-Ethernet2/0/0] ospfv3 1 area 1
[RouterB-Ethernet2/0/0] quit
```

# Configure Router C.

```
[RouterC] ipv6
[RouterC] ospfv3
[RouterC-ospfv3-1] router-id 3.3.3.3
[RouterC-ospfv3-1] quit
[RouterC] interface ethernet 1/0/0
[RouterC-Ethernet1/0/0] ospfv3 1 area 0
[RouterC-Ethernet1/0/0] quit
[RouterC] interface ethernet 2/0/0
[RouterC-Ethernet2/0/0] ospfv3 1 area 2
[RouterC-Ethernet2/0/0] quit
```

# Configure Router D.

```
[RouterD] ipv6
[RouterD] ospfv3
[RouterD-ospfv3-1] router-id 4.4.4.4
[RouterD-ospfv3-1] quit
[RouterD] interface ethernet 1/0/0
[RouterD-Ethernet1/0/0] ospfv3 1 area 2
[RouterD-Ethernet1/0/0] quit
```

# Display the OSPFv3 neighbors of Router B.

```
[RouterB] display ospfv3 peer
OSPFv3 Process (1)
OSPFv3 Area (0.0.0.1)
Neighbor ID Pri State Dead Time Interface Instance ID
1.1.1.1 1 Full/ - 00:00:34 Ethernet2/0/0 0
OSPFv3 Area (0.0.0.0)
Neighbor ID Pri State Dead Time Interface Instance ID
3.3.3.3 1 Full/ - 00:00:32 Ethernet1/0/0 0
```

# Display OSPFv3 neighbors of Router C.

```
[RouterC] display ospfv3 peer
OSPFv3 Process (1)
OSPFv3 Area (0.0.0.0)
Neighbor ID Pri State Dead Time Interface Instance ID
2.2.2.2 1 Full/ - 00:00:37 Ethernet1/0/0 0
OSPFv3 Area (0.0.0.2)
Neighbor ID Pri State Dead Time Interface Instance ID
4.4.4.4 1 Full/ - 00:00:33 Ethernet2/0/0 0
```

# Display the OSPFv3 routing table of Router D.

```
[RouterD] display ospfv3 routing
Codes : E2 - Type 2 External, E1 - Type 1 External, IA - Inter-Area,
N - NSSA, U - Uninstalled
OSPFv3 Process (1)
OSPFv3 Process (1)
 Destination Metric
 Next-hop
IA 1000::/64 2
 via FE80::1572:0:5EF4:1, Ethernet1/0/0
IA 1001::/64 3
 via FE80::1572:0:5EF4:1, Ethernet1/0/0
 1002::/64 1
 directly-connected, Ethernet1/0/0
IA 2000::/64 4
 via FE80::1572:0:5EF4:1, Ethernet1/0/0
```

### Step 3 Configure stub areas.

# Configure the stub area of Router D.

```
[RouterD] ospfv3
[RouterD-ospfv3-1] area 2
[RouterD-ospfv3-1-area-0.0.0.2] stub
[RouterD-ospfv3-1-area-0.0.0.2] quit
```

# Configure the stub area of Router C, and set the cost of the default route advertised to the stub area to 10.

```
[RouterC] ospfv3
[RouterC-ospfv3-1] area 2
[RouterC-ospfv3-1-area-0.0.0.2] stub
[RouterC-ospfv3-1-area-0.0.0.2] default-cost 10
[RouterC-ospfv3-1-area-0.0.0.2] quit
```

# Display the OSPFv3 routing table of Router D, and you can view a new default route in the routing table. Its cost is the sum of the cost of the directly connected routes and the configured cost.

```
[RouterD] display ospfv3 routing
Codes : E2 - Type 2 External, E1 - Type 1 External, IA - Inter-Area,
N - NSSA, U - Uninstalled
OSPFv3 Process (1)
OSPFv3 Process (1)
 Destination Metric
 Next-hop
IA ::/0 11
 via FE80::1572:0:5EF4:1, Ethernet1/0/0
IA 1000::/64 2
 via FE80::1572:0:5EF4:1, Ethernet1/0/0
IA 1001::/64 3
 via FE80::1572:0:5EF4:1, Ethernet1/0/0
 1002::/64 1
 directly-connected, Ethernet1/0/0
IA 2000::/64 4
 via FE80::1572:0:5EF4:1, Ethernet1/0/0
```

### Step 4 Configure totally stub areas.



# Configure Router C and configure Area 2 as a totally stub area.

```
[RouterC] ospfv3
[RouterC-ospfv3-1] area 2
[RouterC-ospfv3-1-area-0.0.0.2] stub no-summary
[RouterC-ospfv3-1-area-0.0.0.2] quit
```

### Step 5 Verify the configuration.

# Display the OSPFv3 routing table of Router D, and you can view that the entries in the routing table decrease; other non-directly connected routes are suppressed; only the default route is reserved.

```
[RouterD] display ospfv3 routing
Codes : E2 - Type 2 External, E1 - Type 1 External, IA - Inter-Area,
N - NSSA, U - Uninstalled
OSPFv3 Process (1)
OSPFv3 Process (1)
 Destination Metric
 Next-hop
 IA ::/0 11
 via FE80::1572:0:5EF4:1, Ethernet1/0/0
 1002::/64 1
 directly-connected, Ethernet1/0/0
```

----End

## Configuration Files

- Configuration file of Router A

```
#
 sysname RouterA
#
 ipv6
#
 interface Ethernet1/0/0
 ipv6 enable
 ipv6 address 2000::1/64
 ospfv3 1 area 0.0.0.1
#
 interface Ethernet2/0/0
 ipv6 enable
 ipv6 address 1001::2/64
 ospfv3 1 area 0.0.0.1
#
 ospfv3 1
 router-id 1.1.1.1
 area 0.0.0.1
#
return
```

- Configuration file of Router B

```
#
 sysname RouterB
#
 ipv6
#
 interface Ethernet1/0/0
 ipv6 enable
 ipv6 address 1000::1/64
 ospfv3 1 area 0.0.0.0
#
 interface Ethernet2/0/0
 ipv6 enable
 ipv6 address 1001::1/64
 ospfv3 1 area 0.0.0.1
#
```

```
ospfv3 1
router-id 2.2.2.2
area 0.0.0.0
area 0.0.0.1
#
return
```

- Configuration file of Router C

```
#
sysname RouterC
#
ipv6
#
interface Ethernet1/0/0
ipv6 enable
ipv6 address 1000::2/64
ospfv3 1 area 0.0.0.0
#
interface Ethernet2/0/0
ipv6 enable
ipv6 address 1002::1/64
ospfv3 1 area 0.0.0.2
#
ospfv3 1
router-id 3.3.3.3
area 0.0.0.0
area 0.0.0.2
stub no-summary
default-cost 10
#
return
```

- Configuration file of Router D

```
#
sysname RouterD
#
ipv6
#
interface Ethernet1/0/0
ipv6 enable
ipv6 address 1002::2/64
ospfv3 1 area 0.0.0.2
#
ospfv3 1
router-id 4.4.4.4
area 0.0.0.2
stub
#
return
```

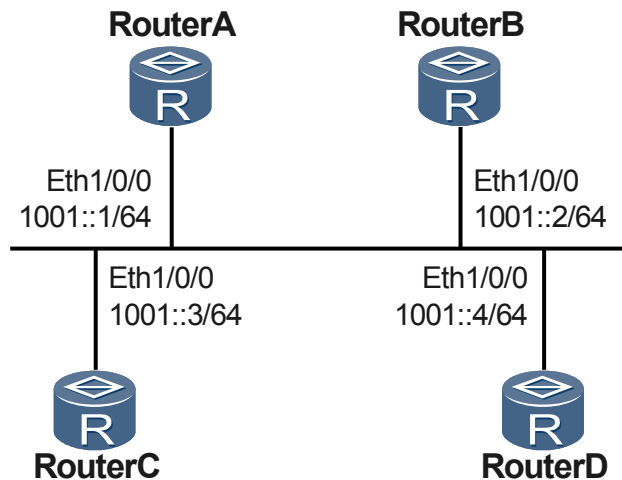
## 6.13.2 Example for Configuring OSPFv3 DR Election

This part provides an example for setting the DR priority on an interface for DR election on a broadcast network.

### Networking Requirements

In [Figure 6-2](#), Router A has a DR priority of 100, which is the highest in the network, so it is elected as the DR. Router C has the second highest priority, so it is elected as the BDR. The priority of Router B is 0 so that it cannot be elected as the DR. Router D does not have a priority and the priority is 1 by default.

Figure 6-2 Networking diagram of configuring OSPFv3 DR election



## Configuration Roadmap

The configuration roadmap is as follows:

1. Configure the router ID on each router, enable OSPFv3, and specify the network segment.
2. Check the DR/BDR status with the default priority.
3. Configure the DR priority on the interface and check the DR/BDR status.

## Data Preparation

To complete the configuration, you need the following data:

- Router A uses the router ID of 1.1.1.1 and DR priority 100.
- Router B uses the router ID of 2.2.2.2 and DR priority 0.
- Router C uses the router ID of 3.3.3.3 and DR priority 2.
- Router D uses the router ID of 4.4.4.4 and default DR priority 1.

## Procedure

**Step 1** Assign an IPv6 address for each interface.

The details are not mentioned here.

**Step 2** Configure basic OSPFv3 functions.

# Configure Router A, enable OSPFv3, and set its router ID to 1.1.1.1.

```
[RouterA] ipv6
[RouterA] ospfv3
[RouterA-ospfv3-1] router-id 1.1.1.1
[RouterA-ospfv3-1] quit
[RouterA] interface ethernet 1/0/0
[RouterA-Ethernet1/0/0] ospfv3 1 area 0
[RouterA-Ethernet1/0/0] quit
```

# Configure Router B, enable OSPFv3, and set its Router ID to 2.2.2.2.

```
[RouterB] ipv6
[RouterB] ospfv3
[RouterB-ospfv3-1] router-id 2.2.2.2
[RouterB-ospfv3-1] quit
[RouterB] interface ethernet 1/0/0
[RouterB-Ethernet1/0/0] ospfv3 1 area 0
[RouterB-Ethernet1/0/0] quit
```

# Configure Router C, enable OSPFv3, and set its Router ID to 3.3.3.3.

```
[RouterC] ipv6
[RouterC] ospfv3
[RouterC-ospfv3-1] router-id 3.3.3.3
[RouterC-ospfv3-1] quit
[RouterC] interface ethernet 1/0/0
[RouterC-Ethernet1/0/0] ospfv3 1 area 0
[RouterC-Ethernet1/0/0] quit
```

# Configure Router D, enable OSPFv3, and set its Router ID to 4.4.4.4.

```
[RouterD] ipv6
[RouterD] ospfv3
[RouterD-ospfv3-1] router-id 4.4.4.4
[RouterD-ospfv3-1] quit
[RouterD] interface ethernet 1/0/0
[RouterD-Ethernet1/0/0] ospfv3 1 area 0
[RouterD-Ethernet1/0/0] quit
```

# Display the neighbors of Router A. You can view the DR priority (its default value is 1) and the neighbor status. Router D is the DR and Router C is the BDR.

#### NOTE

The router with the greater router ID is the DR when routers have the same priority. If a certain Ethernet interface of a router becomes a DR, the other broadcast interfaces of the router have the highest priority in DR election. That is, the DR router is elected as the DR.

```
[RouterA] display ospfv3 peer
OSPFv3 Process (1)
OSPFv3 Area (0.0.0.0)
Neighbor ID Pri State Dead Time Interface Instance ID
2.2.2.2 1 2-Way/DROther 00:00:32 Ethernet1/0/0 0
3.3.3.3 1 Full/Backup 00:00:36 Ethernet1/0/0 0
4.4.4.4 1 Full/DR 00:00:38 Ethernet1/0/0 0
```

# Display the neighbors of Router D, and you can view that all neighbors of Router D are in the Full state.

```
[RouterD] display ospfv3 peer
OSPFv3 Process (1)
OSPFv3 Area (0.0.0.0)
Neighbor ID Pri State Dead Time Interface Instance ID
1.1.1.1 1 Full/DROther 00:00:32 Ethernet1/0/0 0
2.2.2.2 1 Full/DROther 00:00:35 Ethernet1/0/0 0
3.3.3.3 1 Full/Backup 00:00:30 Ethernet1/0/0 0
```

### Step 3 Set the DR priority of the interface.

# Set the DR priority of Router A to 100.

```
[RouterA] interface ethernet 1/0/0
[RouterA-Ethernet1/0/0] ospfv3 dr-priority 100
[RouterA-Ethernet1/0/0] quit
```

# Set the DR priority of Router B to 0.

```
[RouterB] interface ethernet 1/0/0
[RouterB-Ethernet1/0/0] ospfv3 dr-priority 0
[RouterB-Ethernet1/0/0] quit
```

# Set the DR priority of Router C to 2.

```
[RouterC] interface ethernet 1/0/0
[RouterC-Ethernet1/0/0] ospfv3 dr-priority 2
[RouterC-Ethernet1/0/0] quit
```

# Display the neighbors of Router A, and you can view that the DR priority is updated and the DR and BDR remain unchanged.

```
[RouterA] display ospfv3 peer
OSPFv3 Process (1)
OSPFv3 Area (0.0.0.0)
Neighbor ID Pri State Dead Time Interface Instance ID
2.2.2.2 0 2-Way/DROther 00:00:34 Ethernet1/0/0 0
3.3.3.3 2 Full/Backup 00:00:38 Ethernet1/0/0 0
4.4.4.4 1 Full/DR 00:00:31 Ethernet1/0/0 0
```

# Display the neighbors of Router D, and you can view that Router D remains as the DR.

```
[RouterD] display ospfv3 peer
OSPFv3 Process (1)
OSPFv3 Area (0.0.0.0)
Neighbor ID Pri State Dead Time Interface Instance ID
1.1.1.1 100 Full/DROther 00:00:36 Ethernet1/0/0 0
2.2.2.2 0 Full/DROther 00:00:30 Ethernet1/0/0 0
3.3.3.3 2 Full/Backup 00:00:36 Ethernet1/0/0 0
```

#### Step 4 Re-elect the DR/BDR.

# Restart all routers (or run the **shutdown** and **undo shutdown** commands on the interface that establishes the OSPFv3 neighbor relationship), and make OSPFv3 re-elect the DR/BDR.

#### Step 5 Verify the configuration.

# Display the neighbors of Router A, and you can view that Router C is the BDR.

```
[RouterA] display ospfv3 peer
OSPFv3 Process (1)
OSPFv3 Area (0.0.0.0)
Neighbor ID Pri State Dead Time Interface Instance ID
2.2.2.2 0 Full/DROther 00:00:31 Ethernet1/0/0 0
3.3.3.3 2 Full/Backup 00:00:36 Ethernet1/0/0 0
4.4.4.4 1 Full/DROther 00:00:39 Ethernet1/0/0 0
```

# Display the neighbors of Router D, and you can view that Router A is the DR.

```
[RouterD] display ospfv3 peer
OSPFv3 Process (1)
OSPFv3 Area (0.0.0.0)
Neighbor ID Pri State Dead Time Interface Instance ID
1.1.1.1 100 Full/DR 00:00:39 Ethernet1/0/0 0
2.2.2.2 0 2-Way/DROther 00:00:35 Ethernet1/0/0 0
3.3.3.3 2 Full/Backup 00:00:39 Ethernet1/0/0 0
```

----End

## Configuration Files

- Configuration file of Router A

```
#
sysname RouterA
#
ipv6
#
interface Ethernet1/0/0
ipv6 enable
ipv6 address 1001::1/64
```

```
ospfv3 1 area 0.0.0.0
ospfv3 dr-priority 100
#
ospfv3 1
router-id 1.1.1.1
#
return
```

● Configuration file of Router B

```
#
sysname RouterB
#
ipv6
#
interface Ethernet1/0/0
ipv6 enable
ipv6 address 1001::2/64
ospfv3 1 area 0.0.0.0
ospfv3 dr-priority 0
#
ospfv3 1
router-id 2.2.2.2
#
return
```

● Configuration file of Router C

```
#
sysname RouterC
#
ipv6
#
interface Ethernet1/0/0
ipv6 enable
ipv6 address 1001::3/64
ospfv3 1 area 0.0.0.0
ospfv3 dr-priority 2
#
ospfv3 1
router-id 3.3.3.3
#
return
```

● Configuration file of Router D

```
#
sysname RouterD
#
ipv6
#
interface Ethernet1/0/0
ipv6 enable
ipv6 address 1001::4/64
ospfv3 1 area 0.0.0.0
#
ospfv3 1
router-id 4.4.4.4
#
return
```

# 7 IS-IS Configuration

## About This Chapter

This chapter describes the basic principle of IS-IS and procedures for configuring IS-IS, and provides configuration examples.

### [7.1 Basic Concepts of IS-IS](#)

As an IGP, IS-IS is used inside an AS. IS-IS is a link-state protocol. It uses the SPF algorithm to calculate routes.

### [7.2 IS-IS Features Supported by the AR150/200](#)

The AR150/200 supports various Intermediate System-to-Intermediate System (IS-IS) protocol features, including multi-instance, multi-process, hot standby (HSB), multi-topology, local multicast-topology (MT), graceful restart (GR), traffic engineering (TE), DiffServ-aware traffic engineering (DS-TE), administrative tags, Link State Protocol Data Unit (LSP) fragment extension, dynamic host name exchange, fast convergence, Bidirectional Forwarding Detection (BFD), and three-way handshake.

### [7.3 Configuring Basic IPv4 IS-IS Functions](#)

This section describes the procedures for configuring basic IPv4 IS-IS functions, including the procedures for configuring IS-IS processes and interfaces, to implement communication between nodes on an IPv4 IS-IS network.

### [7.4 Establishing or Maintaining IS-IS Neighbor Relationships or Adjacencies](#)

This section describes how to configure the parameters that affect the IS-IS neighbor relationship.

### [7.5 Configuring IPv4 IS-IS Route Selection](#)

Configuring IS-IS route selection can achieve refined control over route selection.

### [7.6 Configuring IPv4 IS-IS Route Summarization](#)

To improve the route searching efficiency and simplify route management on a large-scale IS-IS network, configure IS-IS route summarization to reduce the number of IS-IS routes in a routing table.

### [7.7 Configuring IPv4 IS-IS to Interact with Other Routing Protocols](#)

If other routing protocols are configured on an IS-IS network, you need to configure IS-IS to interact with these protocols to ensure successful communication between them.

### [7.8 Configuring the IPv4 IS-IS Route Convergence Speed](#)

Accelerating IS-IS route convergence can improve the fault location efficiency and improve the network reliability.

#### [7.9 Configuring Static IPv4 BFD for IS-IS](#)

BFD can provide link failure detection featuring light load and high speed (at the millisecond level). Static IPv4 BFD can be configured to monitor IS-IS links.

#### [7.10 Configuring Dynamic IPv4 BFD for IS-IS](#)

Dynamic IPv4 BFD for IS-IS can accelerate IS-IS route convergence.

#### [7.11 Configuring Basic IPv6 IS-IS Functions](#)

This section describes the procedures for configuring basic IPv6 IS-IS functions, including the procedures for configuring IS-IS processes and interfaces, to implement communication between nodes on an IPv6 IS-IS network.

#### [7.12 Configuring IPv6 IS-IS Route Selection](#)

Configuring IS-IS route selection can achieve refined control over route selection.

#### [7.13 Configuring IPv6 IS-IS Route Summarization](#)

To improve the route searching efficiency and simplify route management on a large-scale IS-IS network, configure IS-IS route summarization to reduce the number of IS-IS routes in a routing table.

#### [7.14 Configuring IPv6 IS-IS to Interact with Other Routing Protocols](#)

If other routing protocols are configured on an IS-IS network, you need to configure IS-IS to interact with these protocols to ensure successful communication between them.

#### [7.15 Configuring the IPv6 IS-IS Route Convergence Speed](#)

Accelerating IS-IS route convergence can improve the fault location efficiency and improve the network reliability.

#### [7.16 Configuring IS-IS GR](#)

By configuring IS-IS GR, you can enable Router to restart gracefully and avoid temporary black holes.

#### [7.17 Maintaining IS-IS](#)

Maintaining IS-IS involves resetting IS-IS and clearing IS-IS statistics.

#### [7.18 Configuration Examples](#)

This section provides several configuration examples of IS-IS together with the configuration flowchart. The configuration examples explain networking requirements, configuration notes, and configuration roadmap.



## 7.1 Basic Concepts of IS-IS

As an IGP, IS-IS is used inside an AS. IS-IS is a link-state protocol. It uses the SPF algorithm to calculate routes.

The Intermediate System-to-Intermediate System (IS-IS) is a dynamic routing protocol that was originally created by the International Organization for Standardization (ISO) for its Connectionless Network Protocol (CLNP).

To support the IP routing, the Internet Engineering Task Force (IETF) extended and modified IS-IS in RFC 1195. IS-IS can thus be applied to both TCP/IP and OSI environments. This type of IS-IS is called the Integrated IS-IS or Dual IS-IS.

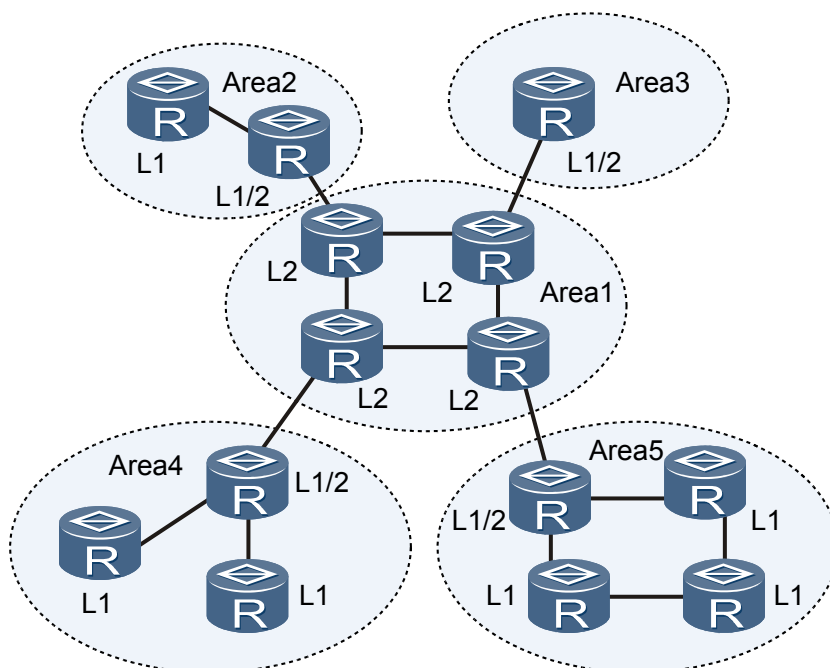
As an Interior Gateway Protocol (IGP), IS-IS is used in Autonomous Systems (ASs). IS-IS is a link-state protocol. It uses the Shortest Path First (SPF) algorithm to calculate routes. It resembles the Open Shortest Path First (OSPF) protocol.

### IS-IS Areas

To support large-scale networks, the IS-IS adopts a two-level structure in a Routing Domain (RD). A large RD is divided into one or more areas. The intra-area routes are managed by the Level-1 routers, whereas the inter-area routes are managed by the Level-2 routers.

**Figure 7-1** shows an IS-IS network. Its topology is similar to that of a multi-area OSPF network. Area 1 is a backbone area. All routers in this area are Level-2 routers. The other four areas are non-backbone areas. They are connected to Area 1 through Level-1-2 routers.

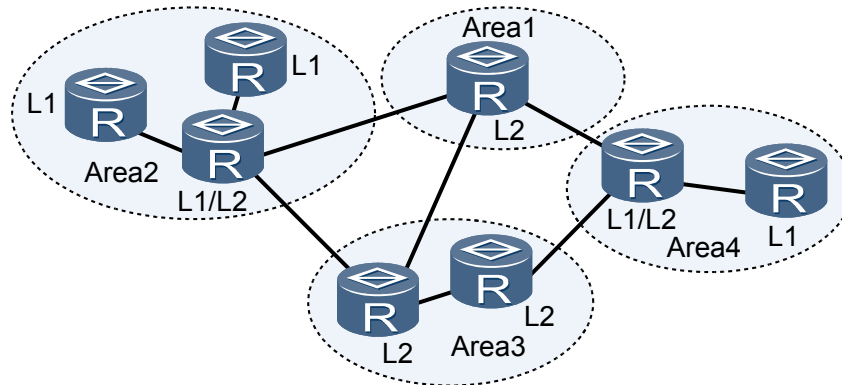
**Figure 7-1** IS-IS topology



**Figure 7-2** shows another type of IS-IS topology. The Level-1-2 routers are used to connect the Level-1 and the Level-2 routers, and are used to establish the backbone network together with

the other Level-2 routers. In this topology, no area is specified as a backbone area. All the Level-2 routers constitute an IS-IS backbone network. The devices may belong to different areas, but the areas must be successive.

Figure 7-2 IS-IS topology II



**NOTE**

The IS-IS backbone network does not refer to a specific area.

This type of networking shows differences between IS-IS and OSPF. For OSPF, the inter-area routes are forwarded by the backbone area and the SPF algorithm is used in the same area. For IS-IS, both Level-1 routers and Level-2 routers use the SPF algorithm to generate Shortest Path Trees (SPTs).

## Network Types

IS-IS supports only two network types, which can be classified as follows according to physical links:

- Broadcast links such as Ethernet and Token-Ring.
- Point-to-point links such as PPP and HDLC.

**NOTE**

For a Non-Broadcast Multi-Access (NBMA) network such as ATM, you need to configure sub-interfaces for it. The type of subnets cannot be Point-to-Multipoint (P2MP). IS-IS cannot run on P2MP networks.

## 7.2 IS-IS Features Supported by the AR150/200

The AR150/200 supports various Intermediate System-to-Intermediate System (IS-IS) protocol features, including multi-instance, multi-process, hot standby (HSB), multi-topology, local multicast-topology (MT), graceful restart (GR), traffic engineering (TE), DiffServ-aware traffic engineering (DS-TE), administrative tags, Link State Protocol Data Unit (LSP) fragment extension, dynamic host name exchange, fast convergence, Bidirectional Forwarding Detection (BFD), and three-way handshake.

 **NOTE**

The IS-IS for IPv6 function is used with a license. To use the IS-IS for IPv6 function, apply for and purchase the following license from the Huawei local office:

- AR150&200 Value-Added Data Package

## Multi-Instance and Multi-Process

IS-IS supports multi-process and multi-instance, facilitating management and improving control efficiency of IS-IS.

- Multi-process

Multi-process allows a group of interfaces to be associated with a specific IS-IS process. This ensures that the specific IS-IS process performs all the protocol-based operations only on the group of interfaces. Multiple IS-IS processes can run on a single router and each process is responsible for a unique group of interfaces.

- Multi-instance

After the VPN feature is enabled, multi-instance allows an IS-IS process to be associated with a specific VPN instance so that all the interfaces of this IS-IS process will be associated with the VPN instance.

## IS-IS GR

GR is a function used to restart a router gracefully. It ensures uninterrupted traffic forwarding during the restart of a router in a short time.

If IS-IS restarts in a non-graceful mode, IS-IS sessions are reset and Link State Protocol Data Units (LSPs) are regenerated and flooded. This triggers the SPF calculation in the entire area, which causes route flapping and forwarding interruption in the area. The IETF defines IS-IS GR in RFC 3847, in which the specifications of protocol restart with FIB tables reserved and unreserved are stated.

 **NOTE**

For details about IS-IS GR, see the "IS-IS" chapter in the *Huawei AR150&200 Series Enterprise Routers Feature Description-IP Routing*.

## Administrative Tag

The use of administrative tags simplifies management. Administrative tags can advertise IP address prefixes in the IS-IS area to control routes. The administrative tag carries the administrative information about an IP address prefix. It is used to control the routes of different levels and routes imported from different areas, various routing protocols, multiple IS-IS instances running on a router, and carrying of tags.

Each administrative tag is associated with certain attributes. If the prefix of the reachable IP address to be advertised by IS-IS has this attribute, IS-IS adds the administrative tag to the reachability TLV in the prefix. In this manner, the tag is advertised throughout the entire IS-IS area.

## LSP Fragment Extension

When more information is carried in an LSP to be advertised by IS-IS, IS-IS advertises multiple LSP fragments. Each LSP fragment is identified by the LSP identifier field of an LSP. The LSP identifier field is 1 byte long. Therefore, the maximum number of fragments that can be generated by an IS-IS router is 256.

The IS-IS fragment extension feature allows an IS-IS router to generate more LSP fragments. To implement this feature, you can use the network manager to configure additional system IDs for the router. Each system ID represents a virtual system, which can generate 256 LSP fragments. With more additional system IDs (up to 50 virtual systems), an IS-IS router can generate a maximum of 13056 LSP fragments.

- Related terms are as follows:
  - Originating system  
In this document, the originating system is the router that actually runs the IS-IS protocol, and each IS-IS process is regarded as multiple virtual routers to generate LSP fragments.
  - Normal system ID  
It is the system ID of the originating system.
  - Additional System-ID  
An additional system ID, assigned by the network administrator, represents a virtual system. Each virtual system is allowed to generate up to 256 extended LSP fragments. Like a normal system ID, an additional system ID must be unique in a routing domain.
  - Virtual system  
It is a virtual system for generating extended LSP fragments. Each virtual system has a unique additional system ID, and each extended LSP fragment carries an additional system ID.
- Operating mode  
An IS-IS router can run the LSP fragment extension feature in the following modes:
  - Mode 1: The originating system sends a link to each virtual system. Then each virtual system sends a link to the originating system. The virtual systems function as the routers that are connected to the originating system on the network. This mode is used when some routers on the network do not support the LSP fragment extension feature. In this mode, only the routing information can be advertised in the LSPs of the virtual systems.
  - Mode 2: All the routers on the network can learn that the LSPs generated by the virtual systems actually belong to the originating system. This mode is used when all the routers on the network support the LSP fragment extension feature. In this mode, all link state information can be advertised in the LSPs of the virtual systems.

## Dynamic Host Name Exchange Mechanism

The dynamic host name exchange mechanism is introduced to conveniently manage and maintain IS-IS networks. The mechanism provides a service of mapping host names to system IDs for the IS-IS routers. The dynamic host name information is advertised in the form of a dynamic host name TLV in an LSP.

The dynamic host name exchange mechanism also provides a service to associate a host name with the designated intermediate system (DIS) on a broadcast network. Then LSPs of pseudo nodes advertise this association in the form of a dynamic host name TLV.

It is easier to identify and memorize the host name than the system ID. After this function is configured, the host name will display when display command is used.

## IS-IS Route Summarization

Route summarization is a function for summarizing routes with the same IP prefix into one route.

On a large-scale IS-IS network, you can configure route summarization to reduce the number of IS-IS routes in the routing table. This improves the usage of system resources and facilitates route management.

IP network segments are not affected when a link frequently alternates between Up and Down on an IP network segment. This prevents route flapping and improves the network stability.

The router supports classless network-based route summarization.

## IS-IS Load Balancing

If there are redundant links on an IS-IS network, there may be multiple equal-cost routes.

Configuring IS-IS load balancing can evenly distribute traffic to each link. This increases the bandwidth usage of each link and prevents network congestion caused by some overloaded links. IS-IS load balancing, however, may affect traffic management because traffic will be randomly forwarded in this mode.

## IS-IS Preference

If there are redundant links on an IS-IS network, there may be multiple equal-cost routes.

The router allows you to configure preference values for equal-cost IS-IS routes so that only the route with the highest preference will be used and the others will function as backups.

This facilitates traffic management, improves the network reliability, and avoids configuration change.

## IS-IS Fast Convergence

- Incremental SPF (I-SPF)

I-SPF calculates only changed routes at a time, but not all routes.

ISO-10589 defines Dijkstra as the algorithm to calculate routes. When a node is added to or removed from a network topology, all routes of all nodes need to be calculated if the Dijkstra algorithm is adopted. As a result, it takes a long time and occupies excessive resources, reducing the route convergence speed of the entire network.

I-SPF improves this algorithm. Except for the first time, only changed nodes instead of all nodes are involved in calculation. The SPT generated at last is the same as that generated by the Dijkstra algorithm. This decreases the CPU usage and speeds up route convergence.

- Partial route calculation (PRC)

Similar to I-SPF, only changed nodes are involved in PRC. PRC, however, does not calculate the shortest path but updates leaf routes based on the SPT calculated by I-SPF.

In route calculation, a leaf represents a route, and a node represents a router. If the SPT calculated using I-SPF changes, PRC calculates all the leaves on only the changed node; if the SPT calculated using I-SPF does not change, PRC calculates only the changed leaf.

For example, if an interface of a node is enabled with IS-IS, the SPT of the entire network remains unchanged. In this case, PRC updates the routes on only the interface of this node, reducing the CPU usage.

PRC working with I-SPF further improves the convergence performance of the network. As an improvement of the original SPF algorithm, PRC and I-SPF replace the original algorithm.

 **NOTE**

In real world applications of AR150/200s, only I-SPF and PRC are used to calculate IS-IS routes.

- **LSP fast flooding**

Based on the RFC, when IS-IS receives LSPs from other routers and the LSPs are more updated than those in its own LSDB, IS-IS uses a timer to flood out the LSPs in the LSDB at specified intervals. Therefore, the LSDB synchronization is slow.

LSP fast flooding addresses the problem. When a router configured with this feature receives one or more LSPs, it floods out the LSPs less than the specified number before route calculation. This accelerates the LSDB synchronization and speeds up network convergence to the great extent.

- **Intelligent timer**

Although the route calculation algorithm is improved, the long interval for triggering route calculation also affects the convergence speed. Using a millisecond timer can shorten the interval, however, excessive CPU resources will be consumed if the network topology changes frequently. An SPF intelligent timer can quickly respond to certain emergent events and also prevent excessive CPU resource consumption.

An IS-IS network running normally is stable. The network seldom changes frequently, and an IS-IS router does not calculate routes frequently. Therefore, you can set a short interval (in milliseconds) for triggering the route calculation for the first time. If the network topology changes frequently, the value of the intelligent timer increases with the calculation times, and the interval for route calculation becomes longer. This prevents excessive CPU resource consumption.

The LSP generation intelligent timer is similar to the SPF intelligent timer. In IS-IS, when the LSP generation timer expires, the system regenerates its own LSP according to the current topology. In the original implementation mechanism, a timer with a fixed value is used, which, however, cannot meet the requirements on fast convergence and low CPU usage. Therefore, the LSP generation timer is designed as an intelligent timer so that it can respond quickly to some emergent events (such as interface alternation between Up and Down) to speed up network convergence. In addition, when the network changes frequently, the value of the intelligent timer becomes greater automatically to prevent excessive CPU resource consumption.

 **NOTE**

Determine whether to configure intelligent timers based on actual network situations and specifications of deployed routers.

## BFD for IS-IS

The AR150/200 supports BFD for IS-IS to detect IS-IS neighbor relationships. BFD can fast detect the faults on links between IS-IS neighbors and reports them to IS-IS. Fast convergence of IS-IS is then implemented.

 **NOTE**

BFD detects only one-hop links between IS-IS neighbors. This is because IS-IS establishes only one-hop neighbors.

- **Static BFD**

To configure static BFD, use command lines to configure single-hop BFD parameters, such as local and remote discriminators. Then configure the device to send BFD session setup requests.

A static BFD session can only be established and released manually. A configuration error will lead to a BFD failure. For example, if the configured local discriminator or remote discriminator is incorrect, a BFD session will not work properly.

The AR150/200 supports static IPv4 BFD for IS-IS.

- **Dynamic BFD**

Dynamic BFD refers to the dynamic establishment of BFD sessions using routing protocols. When a new IS-IS neighbor relationship is set up, BFD is notified of the parameters of the neighbor and the detection parameters (including source and destination IP addresses). Then a BFD session will be established based on the received parameters of the neighbor. Dynamic BFD is more flexible than static BFD.

Connection status between an IS-IS device and its neighbors can be monitored by exchanging Hello packets at intervals. The sending interval is usually set to 10s, and a neighbor is declared Down after at least three intervals (during which no response Hello packet is received from the neighbor). It takes IS-IS some seconds to sense a Down neighbor, resulting in loss of a large amount of high-speed data.

Dynamic BFD can provide link failure detection with light load and high speed (at the millisecond level). Dynamic BFD does not take the place of the Hello mechanism of IS-IS, but helps IS-IS to detect the faults on neighbors or links more quickly, and instruct IS-IS to recalculate routes to correctly guide packet forwarding.

The AR150/200 supports dynamic IPv4 and IPv6 BFD for IS-IS.

 **NOTE**

For details about IS-IS GR, see the "IS-IS" chapter in the *Huawei AR150&200 Series Enterprise Routers Feature Description-IP Routing*.

## IS-IS Three-Way Handshake

A reliable link layer protocol is required when IS-IS runs on a point-to-point (P2P) link. Based on ISO 10589, the two-way handshake mechanism of IS-IS uses Hello packets to set up P2P adjacencies between neighboring routers. Once the router receives a Hello packet from its peer, it regards the status of the peer as Up and sets up an adjacency with the peer.

This mechanism has obvious defects. For example, when an adjacency is set up, the unstable link status causes the loss of Complete Sequence Number Packets (CSNPs). As a result, the LSDB fails to be synchronized during the update period of an LSP. If two or more links exist between two routers, an adjacency can still be set up when one link is Down and the other is Up in the same direction. The parameters of the other link, however, are also used in SPF calculation. Therouter does not detect any fault of the link that is in the Down state and still tries to forward packets over this link.

The three-way handshake mechanism addresses the problem on the unreliable P2P link. In three-way handshake mode, the router regards the neighbor as Up only after confirming that the neighbor receives the packet that it sends and then sets up an adjacency with the neighbor. In addition, a 32-bit circuit ID is used in the three-way handshake mechanism, which is an extension of the local 8-bit circuit ID that defines 255 P2P links.

## 7.3 Configuring Basic IPv4 IS-IS Functions

This section describes the procedures for configuring basic IPv4 IS-IS functions, including the procedures for configuring IS-IS processes and interfaces, to implement communication between nodes on an IPv4 IS-IS network.

## 7.3.1 Establishing the Configuration Task

Before configuring basic IPv4 IS-IS functions, familiarize yourself with the applicable environment, complete the pre-configuration tasks, and obtain the required data.

### Applicable Environment

To deploy IS-IS on an IPv4 network, configure basic IS-IS functions to implement communication between different nodes on the network.

Other IS-IS functions can be configured only after basic IS-IS functions are configured.

Configuring basic IPv4 IS-IS functions includes the following operations:

1. Create IPv4 IS-IS processes.
2. Configure IPv4 IS-IS interfaces.

### Pre-configuration Tasks

Before configuring basic IPv4 IS-IS functions, complete the following tasks:

- Configure a link layer protocol.
- Assign an IP address to each interface to ensure IP connectivity.

### Data Preparation

To configure basic IPv4 IS-IS functions, you need the following data.

| No. | Data                                             |
|-----|--------------------------------------------------|
| 1   | IS-IS process ID                                 |
| 2   | NTE of an IS-IS process                          |
| 3   | Level of each device and level of each interface |

## 7.3.2 Creating IPv4 IS-IS Processes

Before configuring basic IPv4 IS-IS functions, create IPv4 IS-IS processes and then enable IPv4 IS-IS interfaces.

### Context

To create an IPv4 IS-IS process, perform the following operations:

- **Create an IS-IS process and configure the NET of a device.**
- **(Optional) Configure the level of a device.**

The level of a device is **level-1-2** by default.

Configure the device level based on the network planning. If no device level is configured, IS-IS establishes separate neighbor relationships for Level-1 and Level-2 devices and maintains two identical LSDBs, consuming excessive system resources.



- **(Optional) Configure IS-IS host name mapping.**  
After IS-IS host name mapping is configured, a host name but not the system ID of a device will display by using display commands. This configuration improves the maintainability on an IS-IS network.
- **(Optional) Enable the output of the IS-IS adjacency status.**  
If the local terminal monitor is enabled and the output of the IS-IS adjacency status is enabled, IS-IS adjacency changes will be output to the router until the output of the adjacency status is disabled.

## Procedure

- Create an IS-IS process and configure the NET of a device.
  1. Run:  

```
system-view
```

The system view is displayed.
  2. Run:  

```
isis [process-id]
```

An IS-IS process is created, and the IS-IS process view is displayed.

The *process-id* parameter specifies the ID of an IS-IS process. The default value of *process-id* is **1**. To associate an IS-IS process with a VPN instance, run the **isis [ process-id ] [ vpn-instance vpn-instance-name ]** command.
  3. Run:  

```
network-entity net
```

A NET is configured.



### CAUTION

Configuring loopback interface addresses based on NETs is recommended to ensure that a NET is unique on the network. If NETs are not unique, route flapping will easily occur.

Area addresses of NETs are checked when Level-1 IS-IS neighbor relationships are being established, but not checked when Level-2 IS-IS neighbor relationships are being established. Level-1 IS-IS neighbor relationships can be established only if area addresses of NETs are the same.

- (Optional) Configure the level of a device.
  1. Run:  

```
system-view
```

The system view is displayed.
  2. Run:  

```
isis [process-id]
```

An IS-IS process is created, and the IS-IS process view is displayed.
  3. Run:  

```
is-level { level-1 | level-1-2 | level-2 }
```

The level of the router is configured.

- (Optional) Configure IS-IS host name mapping.

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
isis [process-id]
```

An IS-IS process is created, and the IS-IS process view is displayed.

3. Run:

```
is-name symbolic-name
```

IS-IS dynamic host name mapping is configured. The system ID of the local device is mapped to the specified host name.

The value of *symbolic-name* is contained in LSP packets and advertised to other IS-IS devices.

On another IS-IS device displays the value of *symbolic-name*, but not the system ID, of the local IS-IS device.

4. Run:

```
is-name map system-id symbolic-name
```

IS-IS static host name mapping is configured. The system ID of a peer IS-IS device is mapped to the specified host name.

This command configuration takes effect only on the local IS-IS device. The value of *symbolic-name* will not be added to LSP packets.

If dynamic host name mappings is configured on an IS-IS network, the mappings on the network overwrite the mappings configured on the local router.

- (Optional) Enable the output of the IS-IS adjacency status.

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
isis [process-id]
```

An IS-IS process is created, and the IS-IS view is displayed.

3. Run:

```
log-peer-change
```

The output of the adjacency status is enabled.

----End

### 7.3.3 Configuring IPv4 IS-IS Interfaces

To configure an interface on an IS-IS device to send Hello packets or flood LSPs, IS-IS must be enabled on this interface.

## Context

The level of an IS-IS device and level of an interface together determine the level of a neighbor relationship. By default, Level-1 and Level-2 neighbor relationships will be established between two Level-1-2 devices. If only one level of neighbor relationships is required, you can configure the level of an interface to prevent the establishment of the other level of neighbor relationships.

After IS-IS is enabled on an interface, the interface will automatically send Hello packets, attempting to establish neighbor relationships. If a peer device is not an IS-IS device or if an interface is not expected to send Hello packets, suppress the interface. Then this interface only advertises routes of the network segment where the interface reside, but does not send Hello packets. This suppression improves the link bandwidth usage.

## Procedure

- Configure an IS-IS interface.

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
interface interface-type interface-number
```

The interface view is displayed.

3. Run:

```
isis enable [process-id]
```

An IS-IS interface is configured.

After this command is run, the IS-IS device uses the specified interface to send Hello packets and flood LSPs.

### NOTE

No neighbor relationship needs to be established between loopback interfaces. Therefore, if this command is run on a loopback interface, the routes of the network segment where the loopback interface resides will be advertised through other IS-IS interfaces.

- (Optional) Configure the level of an IS-IS interface.

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
interface interface-type interface-number
```

The interface view is displayed.

3. Run:

```
isis circuit-level [level-1 | level-1-2 | level-2]
```

The level of the interface is configured.

By default, the level of an interface is **level-1-2**.

 **NOTE**

Changing the level of an IS-IS interface is valid only when the level of the IS-IS device is Level-1-2. If the level of the IS-IS device is not a Level-1-2, the level of the IS-IS device determines the level of the adjacency to be established.

- (Optional) Suppress an IS-IS interface.

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
interface interface-type interface-number
```

The interface view is displayed.

3. Run:

```
isis silent
```

The IS-IS interface is suppressed.

A suppressed IS-IS interface does not send or receive IS-IS packets. The routes of the network segment where the interface resides, however, can still be advertised to other routers within the area.

---End

## 7.3.4 (Optional) Configuring the IPv4 IS-IS Interfaces

Configuring the IS-IS interface costs can control IS-IS route selection.

### Context

The costs of IS-IS interfaces can be determined in the following modes in descending order by priority:

- Interface cost: is configured for a specified interface.
- Global cost: is configured for all interfaces.
- Automatically calculated cost: is automatically calculated based on the interface bandwidth.

If none of the preceding configurations is performed, the default cost of an IS-IS interface is 10, and the default cost style is narrow.

### Procedure

- Configure the IS-IS cost type.

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
isis [process-id]
```

The IS-IS view is displayed.

3. Run:

```
cost-style { narrow | wide | wide-compatible | { { narrow-compatible | compatible } [relax-spf-limit] } }
```

The IS-IS cost type is configured.

The cost range of an interface and a route received by the interface vary with the cost type.

- If the cost type is narrow, the cost of an interface ranges from 1 to 63. The maximum cost of a route received by the interface is 1023.
- If the cost style is narrow-compatible or compatible, the cost of an interface ranges from 1 to 63. The cost of a received route is related to **relax-spf-limit**.

- If **relax-spf-limit** is not specified, the cost of a route works as follows:

If the cost of a route is not greater than 1023 and the cost of every interface that the route passes through is smaller than or equal to 63, the cost of the route received by the interface is the actual cost.

If the cost of a route is not greater than 1023 but the costs of all interfaces that the route passes through are greater than 63, the IS-IS device can learn only the routes to the network segment where the interface resides and the routes imported by the interface. The cost of the route received by the interface is the actual cost. Subsequent routes forwarded by the interface are discarded.

If the cost of a route is greater than 1023, the IS-IS device can learn only the interface whose route cost exceeds 1023 for the first time. That is, the cost of each interface before this interface is not greater than 63. The routes of the network segment where the interface resides and the routes imported by the interface can all be learned. The cost of the route is 1023. Subsequent routes forwarded by the interface are discarded.

- If **relax-spf-limit** is specified, the cost of a route works as follows:

There is no limit on costs of interfaces or route costs. The cost of a route received by an interface is the actual cost.

- If the cost style is wide-compatible or wide, the cost of the interface ranges from 1 to 16777215. When the cost is 16777215, the neighbor TLV generated on the link cannot be used for route calculation but for the transmission of TE information. The maximum cost of a received route is 0xFFFFFFFF.

- Configure the cost of an IS-IS interface.

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
interface interface-type interface-number
```

The interface view is displayed.

3. Run:

```
isis cost cost [level-1 | level-2]
```

The cost of the IS-IS interface is configured.

You can use the **isis cost** command to configure the cost of a specified interface.

- Configure the global IS-IS cost.

1. Run:

```
system-view
```

The system view is displayed.

2. Run:  
`isis [ process-id ]`
3. Run:  
`circuit-cost cost [ level-1 | level-2 ]`

The IS-IS view is displayed.

The global IS-IS cost is configured.

You can use the **circuit-cost** command to configure the costs of all interfaces at a time.

- Enable IS-IS to automatically calculate interface costs.

1. Run:  
`system-view`
2. Run:  
`isis [ process-id ]`
3. Run:  
`bandwidth-reference value`

The system view is displayed.

The IS-IS view is displayed.

The reference value of the bandwidth is configured. By default, the bandwidth reference value is 100 Mbit/s.

4. Run:  
`auto-cost enable`

The interface is configured to automatically calculate its cost.

The configuration of the bandwidth reference value takes effect only when the cost type is wide or wide-compatible. In this case, Cost of each interface = (Value of **bandwidth-reference**/Interface bandwidth) x 10.

If the cost-style is narrow, narrow-compatible, or compatible, the cost of each interface is based on costs listed in [Table 7-1](#).

**Table 7-1** Mapping between IS-IS interface costs and interface bandwidth

| Cost | Bandwidth Range                               |
|------|-----------------------------------------------|
| 60   | Interface bandwidth ≤ 10 Mbit/s               |
| 50   | 10 Mbit/s < interface bandwidth ≤ 100 Mbit/s  |
| 40   | 100 Mbit/s < interface bandwidth ≤ 155 Mbit/s |
| 30   | 155 Mbit/s < interface bandwidth ≤ 622 Mbit/s |
| 20   | 622 Mbit/s < Interface bandwidth ≤ 2.5 Gbit/s |
| 10   | Interface bandwidth > 2.5 Gbit/s              |

 **NOTE**

To change the cost of a loopback interface, run the **isis cost** command only in the loopback interface view.

----End

## 7.3.5 (Optional) Configuring IPv4 IS-IS Attributes for Interfaces on Different Types of Networks

Different IS-IS attributes can be configured for different types of network interfaces.

### Context

The establishment modes of IS-IS neighbor relationships are different on a broadcast network and on a P2P network. Different IS-IS attributes can be configured for interfaces on different types of networks.

IS-IS is required to select a DIS on a broadcast network. Configure the DIS priorities of IS-IS interfaces so that the interface with the highest priority will be selected as the DIS.

The network types of the IS-IS interfaces on both ends of a link must be the same; otherwise, the IS-IS neighbor relationship cannot be established between the two interfaces. For example, if the type of an interface on a peer device is P2P, you can configure the type of an interface on the local device to P2P so that an IS-IS neighbor relationship can be established between the two devices.

IS-IS on a P2P network is not required to select a DIS. Therefore, you do not need to configure DIS priorities. To ensure the reliability of P2P links, configure IS-IS to use the three-way handshake mode for IS-IS neighbor relationship establishment so that faults on a unidirectional link can be detected.

### Procedure

- Configure the DIS priority of an IS-IS interface.

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
interface interface-type interface-number
```

The interface view is displayed.

3. Run:

```
isis dis-priority priority [level-1 | level-2]
```

The DIS priority is configured on the interface. The greater the value, the higher the priority.

- Configure the network type of an IS-IS interface.

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
interface interface-type interface-number
```

The interface view is displayed.

3. Run:

```
isis circuit-type p2p
```

The network type of the interface is set to P2P.

The network type of an interface is determined by the physical type of the interface by default.

When the network type of an IS-IS interface changes, interface configurations change accordingly.

- After a broadcast interface is configured as a P2P interface using the **isis circuit-type p2p** command, the default settings are restored for the interval for sending Hello packets, the number of Hello packets that IS-IS fails to receive from a neighbor before the neighbor is declared Down, interval for retransmitting LSPs on a P2P link, and various IS-IS authentication modes. Consequently, other configurations such as the DIS priority, DIS name, and interval for sending CSNPs on a broadcast network become invalid.
- After the **undo isis circuit-type** command is run to restore the network type, the default settings are restored for the interval for sending Hello packets, the number of Hello packets that IS-IS fails to receive from a neighbor before the neighbor is declared Down, interval for retransmitting LSPs on a P2P link, various IS-IS authentication modes, DIS priority, and interval for sending CSNPs on a broadcast network.

- Set the negotiation mode in which P2P neighbor relationships can be set up.

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
interface interface-type interface-number
```

The interface view is displayed.

3. Run:

```
isis ppp-negotiation { 2-way | 3-way [only] }
```

The negotiation mode is specified on the interface.

By default, the **3-way** handshake negotiation mode is adopted.

The **isis ppp-negotiation** command can only be used for the establishment of the neighbor relationships on P2P links. In the case of a broadcast link, you can run the **isis circuit-type p2p** command to set the link type to P2P, and then run the **isis ppp-negotiation** command to set the negotiation mode for the establishment of the neighbor relationship.

- Configure OSICP negotiation check on PPP interfaces.

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
interface interface-type interface-number
```



The interface view is displayed.

3. Run:

```
isis ppp-osicp-check
```

The OSICP negotiation status is checked on a PPP interface.

By default, the OSICP negotiation status of a PPP interface does not affect the status of an IS-IS interface.

The **isis ppp-osicp-check** command is applicable only to PPP interfaces. This command is invalid for other P2P interfaces.

After this command is run, the OSICP negotiation status of a PPP interface affects the status of an IS-IS interface. When PPP detects that the OSI network fails, the link status of the IS-IS interface goes Down and the route to the network segment where the interface resides is not advertised through LSPs.

- Configure IS-IS not to check whether the IP addresses of received Hello packets are on the same network segment.

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
interface interface-type interface-number
```

The interface view is displayed.

3. Run:

```
isis peer-ip-ignore
```

IS-IS is configured not to check whether the IP addresses of received Hello packets are on the same network segment.

---End

## 7.3.6 Checking the Configuration

After basic IPv4 IS-IS functions are configured, you can view information about IS-IS neighbors, interfaces, and routes.

### Prerequisites

The configurations of basic IPv4 IS-IS functions are complete.

### Procedure

- Step 1** Run the **display isis name-table** [ *process-id* | **vpn-instance** *vpn-instance-name* ] command to check the mapping from the name of the local device to the system ID.
- Step 2** Run the **display isis peer** [ **verbose** ] [ *process-id* | **vpn-instance** *vpn-instance-name* ] command to check information about IS-IS neighbors.
- Step 3** Run the **display isis interface** [ **verbose** ] [ *process-id* | **vpn-instance** *vpn-instance-name* ] command to check information about IS-IS interfaces.

**Step 4** Run the **display isis route** [ *process-id* | **vpn-instance** *vpn-instance-name* ] [ **ipv4** ] [ **verbose** | [ **level-1** | **level-2** ] | *ip-address* [ *mask* | *mask-length* ] ] \* command to check information about IS-IS routes.

----End

## 7.4 Establishing or Maintaining IS-IS Neighbor Relationships or Adjacencies

This section describes how to configure the parameters that affect the IS-IS neighbor relationship.

### 7.4.1 Establishing the Configuration Task

Before configuring the parameters that affect the IS-IS neighbor relationship, familiarize yourself with the applicable environment, complete the pre-configuration tasks, and obtain the required data. This can help you complete the configuration task quickly and accurately.

#### Applicable Environment

This section describes how to establish or maintain the IS-IS neighbor relationship, covering:

- Adjusting timers of various IS-IS packets, including Hello packets, CSNPs, and LSPs
- Adjusting parameters of LSPs

#### Pre-configuration Tasks

Before establishing or maintaining IS-IS neighbor relationships or adjacencies, complete the following tasks:

- Configuring IP addresses of interfaces to make neighboring nodes reachable
- [7.3 Configuring Basic IPv4 IS-IS Functions](#)

#### Data Preparation

To establish or maintain IS-IS neighbor relationships or adjacencies, you need the following data.

| No. | Data                       |
|-----|----------------------------|
| 1   | Parameters of IS-IS timers |
| 2   | LSP parameters             |

### 7.4.2 Configuring IS-IS Timers for Packets

This part describes how to set the intervals for sending Hello packets, Complete Sequence Number PDUs (CSNPs), and Link State PDUs (LSPs).

## Context

Do as follows on the router that runs IS-IS.

## Procedure

- Configuring the Interval for Sending Hello Packets

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
interface interface-type interface-number
```

The interface view is displayed.

3. Run:

```
isis timer hello hello-interval [level-1 | level-2]
```

The interval for sending the Hello packets is set on an interface.

On a broadcast link, there are Level-1 and Level-2 Hello packets. For different types of packets, you can set different intervals. If no level is specified, both the Level-1 timer and Level-2 timer are configured. On a P2P link, there are only one type of Hello packets. Thus, neither **level-1** nor **level-2** is required.

 **NOTE**

Parameters level-1 and level-2 are configured only on a broadcast interface.

- Configuring the Invalid Number of Hello Packets

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
interface interface-type interface-number
```

The interface view is displayed.

3. Run:

```
isis timer holding-multiplier number [level-1 | level-2]
```

The invalid number of Hello packets is set.

If no level is specified, both the Level-1 timer and Level-2 timer are configured.

 **NOTE**

**level-1** and **level-2** can be found only on the broadcast interface.

IS-IS maintains neighbor relationships with neighbors through Hello packets. If the local router does not receive any Hello packet from a neighbor within holding time, the local router declares that the neighbor is invalid.

In IS-IS, the period during which the local router and its neighbor keep the neighbor relationship is determined by the invalid number of Hello packets and the interval for sending Hello packets.

- Configuring the Interval for Sending CSNPs

1. Run:  
`system-view`  
The system view is displayed.
2. Run:  
`interface interface-type interface-number`  
The interface view is displayed.
3. Run:  
`isis timer csnp csnp-interval [ level-1 | level-2 ]`  
The interval for sending CSNPs is set.

CSNPs are transmitted by the Designated IS (DIS) to synchronize an LSDB in a broadcast network. If the level is not specified, the timer of the current level is configured.

- Configuring the Interval for Retransmitting LSPs

1. Run:  
`system-view`  
The system view is displayed.
2. Run:  
`interface interface-type interface-number`  
The interface view is displayed.
3. Run:  
`isis circuit-type p2p`  
Sets the interface network type as P2P.
4. Run:  
`isis timer lsp-retransmit retransmit-interval`  
The interval for retransmitting LSPs on a P2P link is set.

On a P2P link, if the local router does not receive the response within a period of time after it sends an LSP, it considers that the LSP is lost or dropped. To ensure the reliable transmission, the local router retransmits the LSP according to the *retransmit-interval*. By default, the interval for retransmitting the LSP packet on the P2P link is 5 seconds.

The LSPs sent on a broadcast link do not need any response.

- Configuring the Minimum Interval for Sending LSPs

1. Run:  
`system-view`  
The system view is displayed.
2. Run:  
`interface interface-type interface-number`  
The interface view is displayed.
3. Run:  
`isis timer lsp-throttle throttle-interval [ count count ]`  
The minimum interval for sending LSPs is set.

*count*: specifies the maximum number of LSP packets to be sent within the period specified by *throttle-interval*. The value ranges from 1 to 1000.

You can set the minimum interval for sending LSPs on an IS-IS interface, that is, the delay between two consecutive LSPs. The value is also the interval for sending fragments of a CSNP.

---End

## 7.4.3 Configuring LSP Parameters

By configuring the LSP generation timer, you can adjust the time that an IS-IS network generates LSPs. Setting the size of the LSP to be generated or received by IS-IS can affect the transmission of LSPs.

### Context

Do as follows on the router that runs IS-IS.

### Procedure

- Configuring the Interval for Refreshing LSPs

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
isis [process-id]
```

The IS-IS view is displayed.

3. Run:

```
timer lsp-refresh refresh-time
```

The LSP refreshment period is set.

To synchronize all the LSPs in an area, the routers in the area periodically send all the current LSPs.

By default, the LSP refreshment period is 900 seconds, and the maximum lifetime of an LSP is 1200 seconds. When performing configurations, ensure that the LSP refresh interval is 300 seconds shorter than the maximum LSP Keepalive time. In this way, new LSPs can reach all routers in an area before existing LSPs expire.

#### NOTE

It is recommended to adjust the difference between the LSP refresh period and the maximum Keepalive time of the LSP depending on the network scale.

- Configuring the Max Lifetime of an LSP

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
isis [process-id]
```

The IS-IS view is displayed.

3. Run:  
`timer lsp-max-age age-time`

The lifetime of an LSP is set.

When a router generates an LSP, it sets the max lifetime for the LSP. After the LSP is received by other routers, its lifetime decreases as time passes. If a router does not receive any updated LSP and the lifetime of this LSP decreases to 0, the lifetime of the LSP lasts 60s. If a new LSP is still not received, this LSP is deleted from the LSDB.

- Configuring the Intelligent Timer Used to Generate LSPs

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
isis [process-id]
```

The IS-IS view is displayed.

3. Run:

```
timer lsp-generation max-interval [init-interval [incr-interval]]
[level-1 | level-2]
```

The intelligent timer used to generate LSPs is set.

If no level is configured, both Level-1 and Level-2 are configured.

The initial delay for generating the same LSPs (or LSP fragments) is *init-interval*. The delay for generating the same LSPs (or LSP fragments) secondly is *incr-interval*. When the routes change each time, the delay for generating the same LSPs (or LSP fragments) is twice as the previous value until the delay is up to *max-interval*. After the delay reaches *max-interval* for three times or reset the IS-IS process, the interval is reduced to *init-interval*.

When *incr-interval* is not used and generating the same LSPs (or LSP fragments) for the first time, *init-interval* is used as the initial delay. Then, the delay for generating the same LSPs (or LSP fragments) is *max-interval*. After the delay reaches *max-interval* for three times or the IS-IS process is reset, the interval is reduced to *init-interval*.

When only *max-interval* is used, the intelligent timer changes into a normal one-short timer.

- Configuring the Size of an LSP

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
isis [process-id]
```

The IS-IS view is displayed.

3. Run:

```
lsp-length originate max-size
```

The size of an LSP generated by the system is set.

4. Run:

```
lsp-length receive max-size
```

The size of a received LSP is set.

 **NOTE**

When using *max-size*, ensure that the value of the *max-size* of the generated LSP packet (or the forwarded LSP packet) must be smaller than or equal to that of the received LSP packet.

The value of *max-size* set by using the **lsp-length** command must meet the following conditions.

- The MTU value of an Ethernet interface must be greater than or equal to the sum of *max-size* and 3.
- The MTU value of a P2P interface must be greater than or equal to the value of *max-size*.

- Adding an Interface to a Mesh Group

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
interface interface-type interface-number
```

The VLANIF interface view is displayed.

3. Run:

```
isis mesh-group { mesh-group-number | mesh-blocked }
```

The interface is added to a mesh group.

On the Non Broadcast Multiple Access (NBMA) network, after receiving an LSP, the interface of a router floods the LSP to the other interfaces. In a network with higher connectivity and multiple P2P links, however, the flooding method causes repeated LSP flooding and wastes bandwidth.

To avoid the preceding problem, you can configure several interfaces to form a mesh group. The router in the mesh group does not flood the LSP received from an interface of the group to the other interfaces of the group, but floods it to interfaces of other groups or interfaces that do not belong to any group.

When **mesh-blocked** is configured on an interface, the interface is blocked and cannot flood LSPs outside. All the interfaces added to a mesh group implement global LSDB synchronization through CSNP and PSNP mechanisms.

 **NOTE**

In an ATM or FR network, IS-IS routers are connected through Virtual Circuits (VCs), and the interface here is the logical P2P sub-interface.

- Configuring LSP Fragments Extension

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
isis [process-id]
```

The IS-IS view is displayed.

3. Run:

```
lsp-fragments-extend [[level-1 | level-2 | level-1-2] | [mode-1 | mode-2]] *
```

LSP fragments extension is enabled in an IS-IS process.

4. Run:

```
virtual-system virtual-system-id
```

A virtual system is configured.

To configure a router to generate extended LSP fragments, you must configure at least one virtual system. The ID of the virtual system must be unique in the domain.

An IS-IS process can be configured with up to 50 virtual system IDs.

If neither the mode nor the level is specified when LSP fragments extension is configured, mode-1 and Level-1-2 are used by default.

---End

## 7.4.4 Checking the Configuration

After configuring parameters that affect the IS-IS neighbor relationship, you can check information about the IS-IS interface and statistics about the IS-IS process.

### Prerequisites

The configurations of Establishing or Maintaining IS-IS Neighbor Relationships or Adjacencies are complete.

### Procedure

- Run **display isis interface** [ **verbose** ] [ *process-id* | **vpn-instance** *vpn-instance-name* ] command to check information about the interface enabled with IS-IS.
- Check the statistics of the IS-IS process:
  - **display isis statistics** [ **level-1** | **level-2** | **level-1-2** ] [ *process-id* | **vpn-instance** *vpn-instance-name* ]
  - **display isis statistics packet** [ **interface** *interface-type interface-number* ]
  - **display isis *process-id* statistics** [ **updated-lsp** [ **history** ] ] [ **level-1** | **level-2** | **level-1-2** | **packet** ]

---End

## 7.5 Configuring IPv4 IS-IS Route Selection

Configuring IS-IS route selection can achieve refined control over route selection.

### 7.5.1 Establishing the Configuration Task

Before configuring IPv4 IS-IS route selection, familiarize yourself with the applicable environment, complete the pre-configuration tasks, and obtain the data required for the configuration. This will help you complete the configuration task quickly and efficiently.

### Applicable Environment

After basic IPv4 IS-IS functions are configured, IS-IS routes will be generated, enabling communication between different nodes on a network.



If multiple routes are available, a route discovered by IS-IS may not be the optimal route. This does not meet network planning requirements nor facilitates traffic management. Therefore, configure IPv4 IS-IS route selection to implement refined control over route selection.

To implement refined control over IPv4 IS-IS route selection, perform the following operations:

- **Configuring the IPv4 IS-IS Interfaces.**

 **NOTE**

Changing the IS-IS cost for an interface can achieve the function of controlling route selection, but requires routes on the interface to be recalculated and reconverged when a network topology changes, especially on a large-scale network. In addition, the configuration result may not meet your expectation.

Therefore, the configuration of changing IS-IS costs has best to be finished when configuring basic IS-IS functions.

- Configure IPv4 IS-IS route leaking.
- Configure principles for selecting equal-cost IPv4 IS-IS routes.
- Filter IPv4 IS-IS routes.
- Configure an overload bit for an IPv4 IS-IS device.

## Pre-configuration Tasks

Before configuring IPv4 IS-IS route selection, complete the following tasks:

- Configuring IP addresses for interfaces to ensure that neighboring nodes are reachable at the network layer
- **Configuring Basic IPv4 IS-IS Functions**

## Data Preparation

To configure IPv4 IS-IS route selection, you need the following data.

| No. | Data                                                        |
|-----|-------------------------------------------------------------|
| 1   | ACL for filtering routes, IP prefix list, or routing policy |
| 2   | Maximum number of load-balancing equal-cost IS-IS routes    |
| 3   | Preference of the next hop                                  |
| 4   | Time when an IS-IS device enters the overload state         |

## 7.5.2 Configuring IPv4 IS-IS Route Leaking

Configuring IS-IS route leaking enables you to optimize IS-IS route selection on a two-level-area network.

### Context

If multiple Level-1-2 devices in a Level-1 area are connected to devices in the Level-2 area, a Level-1 LSP sent by each Level-1-2 device carries an ATT flag bit of 1. This Level-1 area will have multiple routes to the Level-2 area and to other Level-1 areas.

By default, routes in a Level-1 area can be leaked into the Level-2 area so that Level-1-2 and Level-2 devices can learn about the topology of the entire network. Devices in a Level-1 area are unaware of the entire network topology because they only maintain LSDBs in the local Level-1 area. Therefore, a device in a Level-1 area can forward traffic to a Level-2 device only through the nearest Level-1-2 device. The route used may not be the optimal route to the destination.

To enable a device in a Level-1 area to select the optimal route, configure IPv4 IS-IS route leaking so that specified routes in the Level-2 area can be leaked into the local Level-1 area.

Routes of services deployed only in the local Level-1 area do not need to be leaked into the Level-2 area. A policy can be configured to leak only desired routes into the Level-2 area.

## Procedure

- Specify routes in the Level-2 area and other Level-1 areas that can be leaked into the local Level-1 area.

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
isis [process-id]
```

The IS-IS view is displayed.

3. Run:

```
import-route isis level-2 into level-1 [tag tag | filter-policy { acl-number | acl-name acl-name | ip-prefix ip-prefix-name | route-policy route-policy-name }] *
```

Routes in the Level-2 area and other Level-1 areas that meet the specified conditions are leaked into the local Level-1 area.

### NOTE

The command is run on the Level-1-2 device that is connected to an external area.

By default, routes in the Level-2 area are not leaked into Level-1 areas. After this command is run, only routes that meet the specified conditions can be leaked into Level-1 areas.

- Configure routes in Level-1 areas to leak into the Level-2 area.

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
isis [process-id]
```

The IS-IS view is displayed.

3. Run:

```
import-route isis level-1 into level-2 [tag tag | filter-policy { acl-number | acl-name acl-name | ip-prefix ip-prefix-name | route-policy route-policy-name }] *
```

Routes that meet the specified conditions in Level-1 areas are leaked into the Level-2 area.

 **NOTE**

The command is run on the Level-1-2 device that is connected to an external area.

By default, all routes in a Level-1 area are leaked into the Level-2 area. After this command is run, only routes that meet the specified conditions can be leaked into the Level-2 area.

----End

## 7.5.3 Configuring Principles for Using Equal-Cost IPv4 IS-IS Routes

If multiple equal-cost IS-IS routes are available on a network, configure the equal-cost IS-IS routes to work in load-balancing mode to increase the bandwidth usage of each link, or configure preference values for the equal-cost IS-IS routes to facilitate traffic management.

### Context

If there are redundant IS-IS links, multiple routes may have an equal cost. Choose either of the following methods to use these equal-cost IS-IS routes:

- Configure load balancing for equal-cost IS-IS routes so that traffic will be evenly balanced among these links.

This mechanism increases the link bandwidth usage and prevents network congestion caused by link overload. However, this mechanism may make traffic management more difficult because traffic will be randomly forwarded.

- Configure preference values for equal-cost IS-IS routes so that only the route with the highest preference will be used and the others function as backups.

This configuration facilitates traffic management and improves the network reliability, without the need to change original configurations.

### Procedure

- Configure equal-cost IS-IS routes to work in load-balancing mode.

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
isis [process-id]
```

The IS-IS view is displayed.

3. Run:

```
maximum load-balancing number
```

The maximum number of load-balancing equal-cost IS-IS routes is set.

 **NOTE**

If the number of IS-IS equal-cost routes is greater than the value of *number*, the number of IS-IS equal-cost routes to work in load-balancing mode is determined by *number*. If the number of IS-IS equal-cost routes is smaller than the value of *number*, IS-IS equal-cost routes of the actual number work in load-balancing mode.

- Configure preference values for equal-cost IS-IS routes.

1. Run:

```
system-view
```

The system view is displayed.

2. Run:  
`isis [ process-id ]`

The IS-IS view is displayed.

3. Run:  
`nexthop ip-address weight value`

A preference value is configured for an equal-cost IS-IS route.

 **NOTE**

A larger value of the *value* parameter indicates a higher preference.

----End

## 7.5.4 Filtering IPv4 IS-IS Routes

If some IS-IS routes are not preferred, configure conditions to filter IS-IS routes. Only IS-IS routes meeting the specified conditions can be added to an IP routing table.

### Context

Only routes in an IP routing table can be used to forward IP packets. An IS-IS route can take effect only after this IS-IS route has been successfully added to an IP routing table.

If an IS-IS route does not need to be added to a routing table, specify conditions, such as a basic ACL, IP prefix, and routing policy, to filter routes so that only IS-IS routes that meet the specified conditions can be added to an IP routing table. IS-IS routes that do not meet the specified conditions cannot be added to the IP routing table and cannot be selected to forward IP packets.

### Procedure

- Step 1** Run:

```
system-view
```

The system view is displayed.

- Step 2** Run:

```
isis [process-id]
```

The IS-IS view is displayed.

- Step 3** Run:

```
filter-policy { acl-number | acl-name acl-name | ip-prefix ip-prefix-name | route-policy route-policy-name } import
```

Conditions for filtering IS-IS routes are configured.

----End

## 7.5.5 Configuring an Overload Bit for an IPv4 IS-IS Device

If an IS-IS device needs to be temporarily isolated, configure the IS-IS device to enter the overload state to prevent other devices from forwarding traffic to this IS-IS device and prevent blackhole routes.

## Context

If an IS (for example, an IS to be upgraded or maintained) needs to be temporarily isolated, configure the IS to enter the overload state so that no device will forward traffic to this IS.

IS-IS routes converge more quickly than BGP routes. To prevent blackhole routes on a network where both IS-IS and BGP are configured, set an overload bit to instruct an IS to enter the overload state during its start or restart. After BGP convergence is complete, cancel the overload bit.

## Procedure

### Step 1 Run:

```
system-view
```

The system view is displayed.

### Step 2 Run:

```
isis [process-id]
```

The IS-IS view is displayed.

### Step 3 Run:

```
set-overload [on-startup [timeout1 | start-from-nbr system-id [timeout1
[timeout2]] | wait-for-bgp [timeout1]]] [allow { interlevel | external }
*]
```

The overload bit is configured.

----End

## 7.5.6 Checking the Configuration

After configuring IPv4 IS-IS route selection, run the following commands to verify that the configurations are correct.

## Procedure

- Run the **display isis route** [ *process-id* | [ **vpn-instance** *vpn-instance-name* ] ] [ **ipv4** ] [ **verbose** | [ **level-1** | **level-2** ] | *ip-address* [ *mask* | *mask-length* ] ] \* [ | **count** ] command to check IS-IS routing information.
- Run the **display isis lsdb** [ { **level-1** | **level-2** } | **verbose** | { **local** | *lsp-id* | **is-name** *symbolic-name* } ] \* [ *process-id* | **vpn-instance** *vpn-instance-name* ] command to check information in the IS-IS LSDB.

----End

## 7.6 Configuring IPv4 IS-IS Route Summarization

To improve the route searching efficiency and simplify route management on a large-scale IS-IS network, configure IS-IS route summarization to reduce the number of IS-IS routes in a routing table.

## Context

Route summarization is used to summarize routes with the same IP prefix into one route.

On a large-scale IS-IS network, route summarization can be configured to reduce the number of IS-IS routes in a routing table. This summarization improves the usage of system resources and facilitates route management.

If a link on an IP network segment that is summarized frequently alternates between Up and Down states, IP network segments that are not summarized will not be affected, preventing route flapping and improving the network stability.

## Procedure

### Step 1 Run:

```
system-view
```

The system view is displayed.

### Step 2 Run:

```
isis [process-id]
```

The IS-IS view is displayed.

### Step 3 Run:

```
summary ip-address mask [avoid-feedback | generate_null0_route | tag tag |
[level-1 | level-1-2 | level-2]] *
```

The specified IS-IS routes are summarized into one IS-IS route.



#### NOTE

After route summarization is configured on an IS, the local routing table still contains all specific routes before the summarization.

The routing tables on other ISs contain only the summary route, and the summary route is deleted only after all its specific routes are deleted.

----End

## Checking the Configuration

After the route summarization function is configured, perform the following steps to check whether the route summarization function has taken effect.

- Run the **display isis route** command to check summary routes in the IS-IS routing table.
- Run the **display ip routing-table [ verbose ]** command to check summary routes in the IP routing table.

## 7.7 Configuring IPv4 IS-IS to Interact with Other Routing Protocols

If other routing protocols are configured on an IS-IS network, you need to configure IS-IS to interact with these protocols to ensure successful communication between them.

### 7.7.1 Establishing the Configuration Task

Before configuring IPv4 IS-IS to interact with other routing protocols, familiarize yourself with the applicable environment, complete the pre-configuration tasks, and obtain the data required for the configuration. This will help you complete the configuration task quickly and efficiently.

## Applicable Environment

If other routing protocols are configured on an IS-IS network, the following issues need to be considered:

- Preference of IS-IS routes

If multiple routes to the same destination are discovered by different routing protocols running on the same device, the route discovered by the protocol with the highest preference is selected. For example, if both OSPF and IS-IS are configured, the route discovered by OSPF is used because OSPF enjoys a higher preference than IS-IS by default.

Therefore, if you want the route discovered by IS-IS to be used, configure IS-IS to have the highest preference.

- Communication between an IS-IS area and other areas

If other routing protocols are configured on an IS-IS network, you need to configure IS-IS to interact with those routing protocols so that IS-IS areas can communicate with non-IS-IS areas.

 **NOTE**

The LSDBs of different IS-IS processes on a device are independent of each other. Therefore, each IS-IS process on the device considers routes of the other IS-IS processes as external routes.

To ensure successful traffic forwarding, configure IS-IS to interact with other routing protocols on a device where external routes are configured, for example, a Level-1-2 IS-IS router. Available methods are as follows:

- Configure IS-IS to advertise a default route.

This mode is easy to configure and does not require devices in IS-IS areas to learn external routes. After a default route is advertised, all traffic in an IS-IS area is forwarded through the default route.

- Configure IS-IS to import external routes.

This mode enables all devices in IS-IS areas to learn external routes, implementing refined control over traffic forwarding.

To ensure successful forwarding of traffic destined for IS-IS areas, you must also enable the other routing protocols to interact with IS-IS.

## Pre-configuration Tasks

Before configuring IPv4 IS-IS to interact with other routing protocols, complete the following tasks:

- Configuring the link layer protocol on interfaces
- Configuring IP addresses for interfaces to ensure that neighboring nodes are reachable at the network layer
- **Configuring Basic IPv4 IS-IS Functions**
- Configuring basic functions of other routing protocols

## Data Preparation

To configure the IPv4 IS-IS route convergence speed, you need the following data.

| No. | Data                                                        |
|-----|-------------------------------------------------------------|
| 1   | ACL for filtering routes, IP prefix list, or routing policy |
| 2   | Preference value of IS-IS                                   |

## 7.7.2 Configuring a Preference Value for IPv4 IS-IS

If multiple routes to the same destination are discovered by different routing protocols, configuring the highest preference value for IS-IS allows a route discovered by IS-IS to be selected preferentially.

### Context

If multiple routes to the same destination are discovered by different routing protocols running on the same device, the route discovered by the protocol with the highest preference is selected.

For example, if both OSPF and IS-IS are configured on a network, the route discovered by OSPF is used because OSPF has a higher preference than IS-IS by default.

To prefer a route discovered by IS-IS, configure a higher preference value for IS-IS. In addition, a routing policy can be configured to increase the preferences of specified IS-IS routes, without affecting route selection.

### Procedure

- Configure the IS-IS preference value.
  1. Run:  
`system-view`  
The system view is displayed.
  2. Run:  
`isis [ process-id ]`  
The IS-IS view is displayed.
  3. Run:  
`preference preference`  
The IS-IS preference value is configured.

#### NOTE

A smaller *preference* value indicates a higher preference.  
The default IS-IS preference value is **15**.

- Configure preference values for specified IS-IS routes.
  1. Run:  
`system-view`  
The system view is displayed.
  2. Run:  
`isis [ process-id ]`



The IS-IS view is displayed.

3. Run:

```
preference preference route-policy route-policy-name
```

The preference values are configured for the specified IS-IS routes.

 **NOTE**

*preference* takes effect only for IS-IS routes that match the specified routing policy.

----End

## 7.7.3 Configuring IPv4 IS-IS to Advertise a Default Route

To forward all traffic in an IS-IS area through a default route, configure IS-IS on a Level-1-2 device to advertise the default route.

### Context

Only the route 0.0.0.0/0 can be advertised as a default route on a Level-1-2 device. All traffic destined for other areas is first forwarded to the Level-1-2 device.

To ensure successful traffic forwarding, external routes must be learned on the Level-1-2 device.

 **NOTE**

Configuring static default routes can also achieve the function of interaction between different routing protocols, but require large configurations and are difficult to manage.

If multiple Level-1-2 devices are deployed, a routing policy can be configured to allow only the Level-1-2 device that meets the specified conditions to advertise a default route, preventing blackhole routes.

### Procedure

**Step 1** Run:

```
system-view
```

The system view is displayed.

**Step 2** Run:

```
isis [process-id]
```

The IS-IS view is displayed.

**Step 3** Run:

```
default-route-advertise [always | match default | route-policy route-policy-name]
[cost cost | tag tag | [level-1 | level-1-2 | level-2]] * [avoid-learning]
```

IS-IS is configured to advertise a default route.

----End

## 7.7.4 Configuring IPv4 IS-IS to Import External Routes

If devices in an IS-IS area need to learn external routes, configure IS-IS on a Level-1-2 device of this area to import external routes.

## Context

If IS-IS is configured on a Level-1-2 device to advertise a default route, all traffic in IS-IS areas will be forwarded by this Level-1-2 device. This will burden this Level-1-2 device because no external route can be learned on the devices in the IS-IS areas.

If multiple Level-1-2 devices are deployed, optimal routes to other areas need to be selected. To ensure optimal routes are selected, all the other devices in the IS-IS areas must learn all or some external routes.

Routing policies can be configured to import or advertise external routes that meet specified conditions to the IS-IS areas.

## Procedure

- Configure IS-IS to import external routes.

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
isis [process-id]
```

The IS-IS view is displayed.

3. Configuring IS-IS to Import External Routes

- If you want to set the cost for the imported route, you can run the **import-route protocol [ process-id ] [ cost-type { external | internal } | cost cost | tag tag | route-policy route-policy-name | [ level-1 | level-2 | level-1-2 ] ] \*** command to import the external routes.
- If you want to keep the original cost for the imported route, you can run the **import-route { { rip | isis | ospf } [ process-id ] | direct | unr | bgp } inherit-cost [ tag tag | route-policy route-policy-name | [ level-1 | level-2 | level-1-2 ] ] \*** command to import the external routes. When configuring IS-IS to retain the original cost value of the imported route, the source routes cannot be **static**.

### NOTE

IS-IS will advertise all imported external routes to the IS-IS areas by default.

If only some imported external routes need to be advertised, run the **filter-policy export** command to set a filtering policy.

- (Optional) Configure IS-IS to advertise some external routes to the IS-IS areas.

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
isis [process-id]
```

The IS-IS view is displayed.

3. Run:

```
filter-policy { acl-number | acl-name acl-name | ip-prefix ip-prefix-name
| route-policy route-policy-name } export [protocol [process-id]]
```

IS-IS is configured to advertise specified external routes to the IS-IS areas.

 **NOTE**

After this command is run, only external routes that meet the specified conditions can be advertised to the IS-IS areas.

----End

## 7.7.5 Checking the Configuration

After IS-IS is enabled to import routes from other protocols, run the following commands to verify that the configurations are correct.

### Procedure

- Run the **display isis lsdb** [ { **level-1** | **level-2** } ] **verbose** [ { **local** | *lsp-id* | **is-name** *symbolic-name* } ] \* [ *process-id* | **vpn-instance** *vpn-instance-name* ] command to check IS-IS LSDB information.
- Run the **display isis route** [ *process-id* | [ **vpn-instance** *vpn-instance-name* ] ] [ **ipv4** ] [ **verbose** | [ **level-1** | **level-2** ] | *ip-address* [ *mask* | *mask-length* ] ] \* [ | **count** ] command to check IS-IS routing information.
- Run the **display ip routing-table ip-prefix** *ip-prefix-name* [ **verbose** ] command to check the IP routing table.

----End

## 7.8 Configuring the IPv4 IS-IS Route Convergence Speed

Accelerating IS-IS route convergence can improve the fault location efficiency and improve the network reliability.

### 7.8.1 Establishing the Configuration Task

Before configuring the IPv4 IS-IS route convergence speed, familiarize yourself with the applicable environment, complete the pre-configuration tasks, and obtain the data required for the configuration. This will help you complete the configuration task quickly and efficiently.

### Applicable Environment

The procedure for implementing IS-IS is as follows:

- Establishment of neighboring relationships: establishes neighboring relationships by exchanging Hello packets between two devices.
- LSP flooding: implements LSDB synchronization between devices in the same area.
- SPF calculation: uses the SPF algorithm to calculate IS-IS routes, and delivers the IS-IS routes to the routing table.

To accelerate the IS-IS route convergence speed, configure the following parameters:

- Interval for detecting IS-IS neighboring device failures
- Flooding parameters of CSNPs and LSPs
- Interval for SPF calculation

You can also configure convergence priorities for IPv4 IS-IS routes so that key routes can be converged by preference when a network topology changes. This minimizes adverse impacts on key services.

## Pre-configuration Tasks

Before configuring the IPv4 IS-IS route convergence speed, complete the following tasks:

- Configuring the link layer protocol on interfaces
- Configuring IP addresses for interfaces to ensure that neighboring nodes are reachable at the network layer
- **Configuring Basic IPv4 IS-IS Functions**

## Data Preparation

To configure the IPv4 IS-IS route convergence speed, you need the following data.

| No. | Data                                                                                 |
|-----|--------------------------------------------------------------------------------------|
| 1   | Interval at which Hello packets are sent and the holding time of neighboring devices |
| 2   | Flooding time of CSNPs and LSPs                                                      |
| 3   | Interval for SPF calculation                                                         |
| 4   | Route convergence priority                                                           |

## 7.8.2 Configuring the Interval for Detecting IS-IS Neighboring Device Failures

To minimize the effects caused by neighboring device failures on an IS-IS network, accelerate the speed of detecting IS-IS neighboring device failures.

### Context

Connection status between an IS-IS device and its neighboring devices can be monitored by exchanging Hello packets at intervals. An IS-IS neighboring device is considered Down if the IS-IS device does not receive any Hello packets from the neighboring device within the specified period (called the holding time). A failure in an IS-IS neighboring device will trigger LSP flooding and SPF calculation, after which IS-IS routes are reconverged.

To speed up fault detection, use the following methods to accelerate the speed of detecting IS-IS neighboring device failures:

- Shorten the interval at which Hello packets are sent.
- Shorten the holding time of neighboring devices.
- **Configuring Dynamic IPv4 BFD for IS-IS.**

 **NOTE**

Configuring IPv4 BFD for IS-IS is recommended because this method provides a faster fault detection speed than the other two methods.

### Procedure

- Set an interval at which Hello packets are sent.

1. Run:  
`system-view`  
The system view is displayed.
2. Run:  
`interface interface-type interface-number`  
The interface view is displayed.
3. Run:  
`isis timer hello hello-interval [ level-1 | level-2 ]`  
The interval at which Hello packets are sent is set.

 **NOTE**

A broadcast link can transmit both Level-1 and Level-2 Hello packets. You can set different sending intervals for these two types of Hello packets. By default, both Level-1 and Level-2 Hello packets are sent.

A P2P link can transmit only one type of Hello packets. Therefore, there is no need to specify the **level-1** or **level-2** parameter if a P2P link is used.

- Set the holding multiplier for neighboring devices.

1. Run:  
`system-view`  
The system view is displayed.
2. Run:  
`interface interface-type interface-number`  
The interface view is displayed.
3. Run:  
`isis timer holding-multiplier number [ level-1 | level-2 ]`  
The holding multiplier of neighboring devices is set.

 **NOTE**

A broadcast link can transmit both Level-1 and Level-2 Hello packets. You can set different sending intervals for these two types of Hello packets. By default, both Level-1 and Level-2 Hello packets are sent.

A P2P link can transmit only one type of Hello packets. Therefore, there is no need to specify the **level-1** or **level-2** parameter if a P2P link is used.

----End

## 7.8.3 Setting Flooding Parameters of SNPs and LSPs

To speed up LSDB synchronization between devices, set flooding parameters of SNPs and LSPs to proper values.

### Context

SNPs consist of CSNPs and PSNPs. CSNPs carry summaries of all LSPs in LSDBs, ensuring LSDB synchronization between neighboring routers. SNPs are processed differently on broadcast links and P2P links.

- On a broadcast link, CSNPs are periodically sent by a DIS device. If a router detects that its LSDB is not synchronized with that on its neighboring router, the router will send PSNPs to apply for missing LSPs.
- On a P2P link, CSNPs are sent only during initial establishment of neighboring relationships. If a request is acknowledged, a neighboring router will send a PSNP in response to a CSNP. If a router detects that its LSDB is not synchronized with that on its neighboring router, the router will also send PSNPs to apply for missing LSPs.

To speed up LSDB synchronization, modify the following parameters of SNPs and LSPs on the AR150/200:

- **Interval at which CSNPs are sent**
- **Intelligent timer controlling LSP generation**
- **Maximum length of LSPs**
- **Refresh interval of LSPs**
- **Maximum lifetime of LSPs**
- **Minimum interval at which LSPs are sent**
- **LSP fast flooding**
- **Interval at which LSPs are retransmitted over a P2P link**

## Procedure

- Set an interval at which CSNPs are sent.
  1. Run:  

```
system-view
```

The system view is displayed.
  2. Run:  

```
interface interface-type interface-number
```

The interface view is displayed.
  3. Run:  

```
isis timer csnp csnp-interval [level-1 | level-2]
```

The interval at which CSNPs are sent is set on the specified interface.

### NOTE

Configure **Level-1** and **Level-2** only when a broadcast interface is specified.

- Configure the intelligent timer controlling LSP generation.
  1. Run:  

```
system-view
```

The system view is displayed.
  2. Run:  

```
isis [process-id]
```

The IS-IS view is displayed.
  3. Run:  

```
timer lsp-generation max-interval [init-interval [incr-interval]]
[level-1 | level-2]
```

The intelligent timer controlling LSP generation is configured.

If a level is not specified, both **level-1** and **level-2** are used by default.

The delay in generating an LSP or an LSP fragment for the first time is determined by *init-interval*; the delay in generating an LSP or an LSP fragment for the second time is determined by *incr-interval*. From the third time on, the delay in generating an LSP increases twice every time until the delay reaches the value specified by *max-interval*. After the delay remains at the value specified by *max-interval* for three times or the IS-IS process is restarted, the delay decreases to the value specified by *init-interval*.

If *incr-interval* is not specified, the delay in generating an LSP or LSP fragment for the first time is determined by *init-interval*. From the second time on, the delay in generating an LSP is determined by *max-interval*. After the delay remains at the value specified by *max-interval* for three times or the IS-IS process is restarted, the delay decreases to the value specified by *init-interval*.

When only *max-interval* is specified, the intelligent timer functions as an ordinary one-time triggering timer.

- Set the maximum length for LSPs.

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
isis [process-id]
```

The IS-IS view is displayed.

3. Run:

```
lsp-length originate max-size
```

The maximum length is set for each LSP to be generated.

4. Run:

```
lsp-length receive max-size
```

The maximum length is set for each LSP to be received.

 **NOTE**

Ensure that the value of *max-size* for LSPs to be generated must be smaller than or equal to the value of *max-size* for LSPs to be received.

The value of *max-size* in the **lsp-length** command must meet the following conditions.

- The MTU of an Ethernet interface must be greater than or equal to the sum of the value of *max-size* and 3.
- The MTU of a P2P interface must be greater than or equal to the value of *max-size*.

- Set the refresh interval for LSPs.

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
isis [process-id]
```

The IS-IS view is displayed.

3. Run:

```
timer lsp-refresh refresh-time
```

A refresh interval is set for LSPs.

To synchronize all LSPs in the areas, IS-IS regularly transmits all the current LSPs to neighbors.

By default, the LSP refresh interval is 900s, and the maximum lifetime of an LSP is 1200s. Ensure that the LSP refresh interval is more than 300s shorter than the maximum LSP lifetime. This allows new LSPs to reach all routers in an area before existing LSPs expire.

 **NOTE**

The larger a network, the greater the deviation between the LSP refresh interval and the maximum LSP lifetime.

● Set the maximum lifetime for LSPs.

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
isis [process-id]
```

The IS-IS view is displayed.

3. Run:

```
timer lsp-max-age age-time
```

The maximum lifetime is set for LSPs.

When a router generates the system LSP, it fills in the maximum lifetime for this LSP. After this LSP is received by other routers, the lifetime of the LSP is reduced gradually. If the router does not receive any more update LSPs and the lifetime of the LSP is reduced to 0, the LSP will be deleted from the LSDB 60s later if no more updated LSPs are received.

● Set the minimum interval at which LSPs are sent.

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
interface interface-type interface-number
```

The interface view is displayed.

3. Run:

```
isis timer lsp-throttle throttle-interval [count count]
```

The minimum interval at which LSPs are sent is set.

The *count* parameter specifies the maximum number of LSPs that can be sent within the interval specified by *throttle-interval*. The value of *count* is an integer ranging from 1 to 1000.

● Enable LSP fast flooding.



1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
isis [process-id]
```

The IS-IS view is displayed.

3. Run:

```
flash-flood [lsp-count | max-timer-interval interval | [level-1 |
level-2]] *
```

The LSP fast flooding is enabled.

Running the **flash-flood** command speeds up LSP flooding. The *lsp-count* parameter specifies the number of LSPs flooded each time, which is applicable to all interfaces. If the number of LSPs to be sent is greater than the value of *lsp-count*, *lsp-count* takes effect. If the number of LSPs to be sent is smaller than the value of *lsp-count*, LSPs of the actual number are sent. If a timer is configured and the configured timer does not expire before the route calculation, the LSPs are flooded immediately when being received; otherwise, the LSPs are sent when the timer expires.

When LSP fast flooding is enabled, Level-1 LSPs and Level-2 LSPs are fast flooded by default if no level is specified.

- Set an interval at which LSPs are retransmitted over a P2P link.

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
interface interface-type interface-number
```

The interface view is displayed.

3. (Optional) Run:

```
isis circuit-type p2p
```

Emulate a broadcast interface to the P2P interface.

4. Run:

```
isis timer lsp-retransmit retransmit-interval
```

The interval at which LSPs are retransmitted over a P2P link is set.

---End

## 7.8.4 Setting the SPF Calculation Interval

To improve the fault location efficiency on an IS-IS network and prevent SPF calculation from consuming excessive system resources, set the SPF calculation interval to a proper value.

### Context

A network change always triggers IS-IS to perform SPF calculation. Frequent SPF calculation will consume excessive CPU resources, affecting services.

To solve this problem, configure an intelligent timer to control the interval for SPF calculation. For example, to speed up IS-IS route convergence, set the interval for SPF calculation to a small value, and set the interval to a large value after the IS-IS network becomes stable.

## Procedure

### Step 1 Run:

```
system-view
```

The system view is displayed.

### Step 2 Run:

```
isis [process-id]
```

The IS-IS view is displayed.

### Step 3 Run:

```
timer spf max-interval [init-interval [incr-interval]]
```

The SPF intelligent timer is configured.

The intelligent timer changes as follows:

- The delay for the first SPF calculation is determined by *init-interval*; the delay for the second SPF calculation is determined by *incr-interval*. From the third time on, the delay in SPF calculation increases twice every time until the delay reaches the value specified by *max-interval*. After the delay remains at the value specified by *max-interval* for three times or the IS-IS process is restarted, the delay decreases to the value specified by *init-interval*.
- If *incr-interval* is not specified, the delay in SPF calculation for the first time is determined by *init-interval*. From the second time on, the delay in SPF calculation is determined by *max-interval*. After the delay remains at the value specified by *max-interval* for three times or the IS-IS process is restarted, the delay decreases to the value specified by *init-interval*.
- When only *max-interval* is specified, the intelligent timer functions as an ordinary one-time triggering timer.

---End

## 7.8.5 Configuring Convergence Priorities for IPv4 IS-IS Routes

If some IS-IS routes need to be converged by preference to minimize adverse impacts on services, configure those routes to have the highest convergence priority.

## Context

By default, the convergence priority of 32-bit host routes is **medium**, and the convergence priority of the other IS-IS routes is **low**.

The AR150/200 allows you to configure the highest convergence priority for specific IS-IS routes so that those IS-IS routes will be converged first when a network topology changes.

## Procedure

### Step 1 Run:

```
system-view
```

The system view is displayed.

**Step 2** Run:

```
isis [process-id]
```

The IS-IS view is displayed.

**Step 3** Run:

```
prefix-priority [level-1 | level-2] { critical | high | medium } { ip-prefix
prefix-name | tag tag-value }
```

Convergence priorities are set for IS-IS routes.

The application rules of the convergence priorities for IS-IS routes are as follows:

- Existing IS-IS routes are converged based on the priorities configured in the **prefix-priority** command.
- New IS-IS routes are converged based on the priorities configured in the **prefix-priority** command.
- If an IS-IS route conforms to the matching rules of multiple convergence priorities, the highest convergence priority is used.
- The convergence priority of a Level-1 IS-IS route is higher than that of a Level-2 IS-IS route.
- If the route level is not specified, the configuration of the **prefix-priority** command takes effect for both Level-1 and Level-2 IS-IS routes.

 **NOTE**

The **prefix-priority** command is only applicable to the public network.

After the **prefix-priority** command is run, the convergence priority of 32-bit host routes is **low**, and the convergence priorities of the other routes are determined as specified in the **prefix-priority** command.

**Step 4** (Optional) Run:

```
quit
```

The system view is displayed.

**Step 5** (Optional) Run:

```
ip route prefix-priority-scheduler critical-weight high-weight medium-weight low-
weight
```

The preference-based scheduling ratio of IPv4 routes is configured.

By default, the preference-based scheduling ratio of IPv4 routes is 8:4:2:1.

----End

## 7.8.6 Checking the Configuration

After the parameters specifying the IPv4 IS-IS route convergence speed are set, run the following commands to verify that the configurations are correct.

### Procedure

- Run the **display isis interface [ verbose ] [ process-id | vpn-instance vpn-instance-name ]** command to check IS-IS packet information.

- Run the **display isis route** [ *process-id* | **vpn-instance** *vpn-instance-name* ] [ **ipv4** ] [ **verbose** | [ **level-1** | **level-2** ] | *ip-address* [ *mask* | *mask-length* ] ] \* [ | **count** ] command to check the preference of IS-IS routes.

----End

## 7.9 Configuring Static IPv4 BFD for IS-IS

BFD can provide link failure detection featuring light load and high speed (at the millisecond level). Static IPv4 BFD can be configured to monitor IS-IS links.

### Context

In a static BFD session scenario, you need to configure single-hop BFD parameters, such as local and remote discriminators and then configure the device to send BFD session setup requests.

A static BFD session can only be established and released manually. A configuration error will lead to a BFD failure. For example, if a local or remote discriminator is incorrectly configured, a BFD session will not work properly.

#### NOTE

A BFD session currently does not detect route switching. If the change of bound peer IP address causes a route to switch to another link, the BFD session is negotiated again only when the original link fails.

### Pre-configuration Tasks

Before configuring static IPv4 BFD for IS-IS, complete the following tasks:

- Assign an IP address to each interface to ensure IP connectivity.
- [Configuring Basic IPv4 IS-IS Functions](#)

### Configuration Roadmap

The configuration roadmap is as follows:

| No. | Data                                                    |
|-----|---------------------------------------------------------|
| 1   | Type and number of the interface to be enabled with BFD |

### Procedure

- Enable BFD globally.
  1. Run:  
**system-view**  
  
The system view is displayed.
  2. Run:  
**bfd**  
  
BFD is enabled globally.

3. Run:  
`quit`

The system view is displayed.

- Configure a single-hop BFD session.

1. Run:

```
bfd cfg-name bind peer-ip ip-address [interface interface-type interface-number]
```

BFD is enabled between the specified interface and peer router.

If a peer IP address and a local interface are specified in the **bfd** command, BFD monitors only a single-hop link with the interface specified in the **bfd** command as the outbound interface and with the peer IP address specified in the **peer-ip** command as the next-hop address.

2. Set discriminators.

- Run:

```
discriminator local discr-value
```

A local discriminator is set.

- Run:

```
discriminator remote discr-value
```

A remote discriminator is set.

The local discriminator of a device must be the remote discriminator of the device on the other end; otherwise, a BFD session cannot be established. In addition, the local and remote discriminators cannot be modified after being configured.

 **NOTE**

The local discriminator set using the **local** *discr-value* command on a device must be the same as the remote discriminator set using the **remote** *discr-value* command on the device of the other end.

3. Run:  
`commit`

Configurations are committed.

4. Run:  
`quit`

The system view is displayed.

- Enable static IPv4 BFD on an interface.

1. Run:

```
interface interface-type interface-number
```

The view of the specified interface is displayed.

2. Run:

```
isis bfd static
```

Static IPv4 BFD is enabled on the specified interface.

----End

## Checking the Configuration

Information about a BFD session can be viewed only after parameters of the BFD session are set and the BFD session is established.

Run the **display isis interface verbose** command. The command output shows that the status of static BFD for IS-IS process 1 is Yes.

## 7.10 Configuring Dynamic IPv4 BFD for IS-IS

Dynamic IPv4 BFD for IS-IS can accelerate IS-IS route convergence.

### Context

Connection status between an IS-IS device and its neighbors can be monitored by exchanging Hello packets at intervals. The minimum allowable sending interval is 3s, and a neighbor is declared Down after at least three intervals during which no response Hello packet is received from the neighbor. IS-IS takes more than one second to detect that a neighbor becomes Down, resulting in loss of a large amount of high-speed data.

To solve this problem, BFD must be configured for IS-IS. IPv4 BFD provides millisecond-level fault detection. After detecting a link or node failure, BFD will notify IS-IS of the failure, accelerating the IS-IS route convergence speed.

Dynamic IPv4 BFD for IS-IS implements dynamic setup of BFD sessions. When a new IS-IS neighbor relationship is set up, BFD is notified of the neighbor parameters and the detection parameters (including source and destination IP addresses). Then a BFD session will be established based on the received neighbor parameters. Dynamic BFD is more flexible than static BFD.

#### NOTE

A BFD session currently does not detect route switching. If the change of bound peer IP address causes a route to switch to another link, the BFD session is negotiated again only when the original link fails.

### Pre-configuration Tasks

Before configuring dynamic IPv4 BFD for IS-IS, complete the following tasks:

- Assign an IP address to each interface to ensure IP connectivity.
- [Configuring Basic IS-IS Functions](#)

### Configuration Roadmap

The configuration roadmap is as follows:

| No. | Data                                                    |
|-----|---------------------------------------------------------|
| 1   | Number of the IS-IS process to be enabled with BFD      |
| 2   | Type and number of the interface to be enabled with BFD |
| 3   | Parameter values of a BFD session                       |

You can use either of the following methods to enable dynamic IPv4 BFD for IS-IS:

- **Enable dynamic IPv4 BFD for specified IS-IS processes.** This method is recommended if you need to enable dynamic IPv4 BFD for IS-IS on a large number of IS-IS interfaces.
- **Enable dynamic IPv4 BFD for specified interfaces.** This method is recommended if you need to enable dynamic IPv4 BFD for IS-IS on a small number of IS-IS interfaces.

## Procedure

- Enable dynamic IPv4 BFD for an IS-IS process.
  1. Run:  

```
system-view
```

The system view is displayed.
  2. Run:  

```
bfd
```

BFD is enabled globally.
  3. Run:  

```
quit
```

The system view is displayed.
  4. Run:  

```
isis process-id
```

The IS-IS view is displayed.
  5. Run:  

```
bfd all-interfaces enable
```

BFD for IS-IS is enabled.

After BFD is enabled globally and the neighbor status becomes Up, IS-IS adopts default BFD parameters to establish BFD sessions on all interfaces.
  6. (Optional) Run:  

```
bfd all-interfaces { min-rx-interval receive-interval | min-tx-interval transmit-interval | detect-multiplier multiplier-value } *
```

The parameters for establishing BFD sessions are set for all interfaces.

The command execution result is applicable to BFD session parameters on all IS-IS interfaces.
  7. Run:  

```
quit
```

The system view is displayed.

To disable the BFD function on an interface, run the **isis bfd block** command in the interface view to disable the interface from establishing BFD sessions.
- Enable dynamic IPv4 BFD on an interface.
  1. Run:  

```
system-view
```

The system view is displayed.
  2. Run:  

```
bfd
```

BFD is enabled globally.

3. Run:  
`quit`

The system view is displayed.

4. Run:  
`interface interface-type interface-number`

The interface view is displayed.

5. Run:  
`isis bfd enable`

BFD is enabled on the interface.

After BFD is configured globally and the neighbor status is Up (on a broadcast network, DIS is in the Up state), default BFD parameters will be used to establish BFD sessions on the specified interface.

6. (Optional) Run:  
`isis bfd { min-rx-interval receive-interval | min-tx-interval transmit-interval | detect-multiplier multiplier-value } *`

Run this command when BFD session parameters need to be configured for a specified interface.

#### NOTE

The priority of BFD configured on an interface is higher than that of BFD configured for a process. If BFD session parameters are configured for both a process and an interface, the parameters on the interface will be used to establish a dynamic BFD session.

----End

## Checking the Configuration

After BFD is enabled on both ends of a link, run the `display isis [ process-id | vpn-instance vpn-instance-name ] bfd session { all | peer ip-address | interface interface-type interface-number }` command.

## 7.11 Configuring Basic IPv6 IS-IS Functions

This section describes the procedures for configuring basic IPv6 IS-IS functions, including the procedures for configuring IS-IS processes and interfaces, to implement communication between nodes on an IPv6 IS-IS network.

### 7.11.1 Establishing the Configuration Task

Before configuring basic IPv6 IS-IS functions, familiarize yourself with the applicable environment, complete the pre-configuration tasks, and obtain the required data.

#### Applicable Environment

To deploy IS-IS on an IPv6 network, configure basic IS-IS functions to implement communication between different nodes on the network.

Other IS-IS functions can be configured only after basic IS-IS functions are configured.



Configuring basic IPv6 IS-IS functions includes the following operations:

1. Create IPv6 IS-IS processes.
2. Configure IPv6 IS-IS interfaces.

## Pre-configuration Tasks

Before configuring basic IPv6 IS-IS functions, complete the following tasks:

- Configure a link layer protocol.
- Assign an IPv6 address to each interface to ensure IP connectivity.
- Enable the IPv6 in system view.

## Data Preparation

To configure basic IPv6 IS-IS functions, you need the following data.

| No. | Data                                             |
|-----|--------------------------------------------------|
| 1   | IS-IS process ID                                 |
| 2   | NTE of an IS-IS process                          |
| 3   | Level of each device and level of each interface |

### 7.11.2 Creating IPv6 IS-IS Processes

Before configuring basic IPv6 IS-IS functions, create IPv6 IS-IS processes and then enable IPv6 IS-IS interfaces.

#### Context

To create an IPv6 IS-IS process, perform the following operations:

- **Create an IS-IS process and configure the NET of a device.**
- **(Optional) Configure the level of a device.**

The level of a device is **level-1-2** by default.

Configure the device level based on the network planning. If no device level is configured, IS-IS establishes separate neighbor relationships for Level-1 and Level-2 devices and maintains two identical LSDBs, consuming excessive system resources.

- **(Optional) Configure IS-IS host name mapping.**

After IS-IS host name mapping is configured, a host name but not the system ID of a device will display by using display commands. This configuration improves the maintainability on an IS-IS network.

- **(Optional) Enable the output of the IS-IS adjacency status.**

If the local terminal monitor is enabled and the output of the IS-IS adjacency status is enabled, IS-IS adjacency changes will be output to the router until the output of the adjacency status is disabled.

## Procedure

- Create an IS-IS process and configure the NET of a device, enable IPv6 for the process.

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
isis [process-id]
```

An IS-IS process is created, and the IS-IS process view is displayed.

The *process-id* parameter specifies the ID of an IS-IS process. The default value of *process-id* is **1**.

3. Run:

```
network-entity net
```

A NET is configured.



### CAUTION

Configuring loopback interface addresses based on NETs is recommended to ensure that a NET is unique on the network. If NETs are not unique, route flapping will easily occur.

Area addresses of NETs are checked when Level-1 IS-IS neighbor relationships are being established, but not checked when Level-2 IS-IS neighbor relationships are being established. Level-1 IS-IS neighbor relationships can be established only if area addresses of NETs are the same.

4. Run:

```
ipv6 enable
```

The IPv6 of IS-IS process is enabled.

- (Optional) Configure the level of a device.

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
isis [process-id]
```

An IS-IS process is created, and the IS-IS process view is displayed.

3. Run:

```
is-level { level-1 | level-1-2 | level-2 }
```

The level of the router is configured.

- (Optional) Configure IS-IS host name mapping.

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
isis [process-id]
```

An IS-IS process is created, and the IS-IS process view is displayed.

3. Run:

```
is-name symbolic-name
```

IS-IS dynamic host name mapping is configured. The system ID of the local device is mapped to the specified host name.

The value of *symbolic-name* is contained in LSP packets and advertised to other IS-IS devices.

On another IS-IS device displays the value of *symbolic-name*, but not the system ID, of the local IS-IS device.

4. Run:

```
is-name map system-id symbolic-name
```

IS-IS static host name mapping is configured. The system ID of a peer IS-IS device is mapped to the specified host name.

This command configuration takes effect only on the local IS-IS device. The value of *symbolic-name* will not be added to LSP packets.

If dynamic host name mappings is configured on an IS-IS network, the mappings on the network overwrite the mappings configured on the local router.

- (Optional) Enable the output of the IS-IS adjacency status.

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
isis [process-id]
```

An IS-IS process is created, and the IS-IS view is displayed.

3. Run:

```
log-peer-change
```

The output of the adjacency status is enabled.

----End

## 7.11.3 Configuring IPv6 IS-IS Interfaces

To configure an interface on an IS-IS device to send Hello packets or flood LSPs, IS-IS must be enabled on this interface.

### Context

The level of an IS-IS device and level of an interface together determine the level of a neighbor relationship. By default, Level-1 and Level-2 neighbor relationships will be established between two Level-1-2 devices. If only one level of neighbor relationships is required, you can configure the level of an interface to prevent the establishment of the other level of neighbor relationships.

After IS-IS is enabled on an interface, the interface will automatically send Hello packets, attempting to establish neighbor relationships. If a peer device is not an IS-IS device or if an

interface is not expected to send Hello packets, suppress the interface. Then this interface only advertises routes of the network segment where the interface reside, but does not send Hello packets. This suppression improves the link bandwidth usage.

## Procedure

- Configure an IS-IS interface.

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
interface interface-type interface-number
```

The interface view is displayed.

3. Run:

```
ipv6 enable
```

The IPv6 of interface is enabled.

4. Run:

```
isis ipv6 enable [process-id]
```

An IS-IS interface is configured.

After this command is run, the IS-IS device uses the specified interface to send Hello packets and flood LSPs.

### NOTE

No neighbor relationship needs to be established between loopback interfaces. Therefore, if this command is run on a loopback interface, the routes of the network segment where the loopback interface resides will be advertised through other IS-IS interfaces.

- (Optional) Configure the level of an IS-IS interface.

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
interface interface-type interface-number
```

The interface view is displayed.

3. Run:

```
isis circuit-level [level1-1 | level1-1-2 | level-2]
```

The level of the interface is configured.

By default, the level of an interface is **level-1-2**.

### NOTE

Changing the level of an IS-IS interface is valid only when the level of the IS-IS device is Level-1-2. If the level of the IS-IS device is not a Level-1-2, the level of the IS-IS device determines the level of the adjacency to be established.

- (Optional) Suppress an IS-IS interface.

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
interface interface-type interface-number
```

The interface view is displayed.

3. Run:

```
isis silent
```

The IS-IS interface is suppressed.

A suppressed IS-IS interface does not send or receive IS-IS packets. The routes of the network segment where the interface resides, however, can still be advertised to other routers within the area.

----End

## 7.11.4 (Optional) Configuring the IPv6 IS-IS Interfaces

Configuring the IS-IS interface costs can control IS-IS route selection.

### Context

The costs of IS-IS interfaces can be determined in the following modes in descending order by priority:

- Interface cost: is configured for a specified interface.
- Global cost: is configured for all interfaces.
- Automatically calculated cost: is automatically calculated based on the interface bandwidth.

If none of the preceding configurations is performed, the default cost of an IS-IS interface is 10, and the default cost style is narrow.

### Procedure

- Configure the IS-IS cost type.

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
isis [process-id]
```

The IS-IS view is displayed.

3. Run:

```
cost-style { narrow | wide | wide-compatible | { { narrow-compatible | compatible } [relax-spf-limit] } }
```

The IS-IS cost type is configured.

The cost range of an interface and a route received by the interface vary with the cost type.

- If the cost type is narrow, the cost of an interface ranges from 1 to 63. The maximum cost of a route received by the interface is 1023.
- If the cost style is narrow-compatible or compatible, the cost of an interface ranges from 1 to 63. The cost of a received route is related to **relax-spf-limit**.

- If **relax-spf-limit** is not specified, the cost of a route works as follows:
    - If the cost of a route is not greater than 1023 and the cost of every interface that the route passes through is smaller than or equal to 63, the cost of the route received by the interface is the actual cost.
    - If the cost of a route is not greater than 1023 but the costs of all interfaces that the route passes through are greater than 63, the IS-IS device can learn only the routes to the network segment where the interface resides and the routes imported by the interface. The cost of the route received by the interface is the actual cost. Subsequent routes forwarded by the interface are discarded.
    - If the cost of a route is greater than 1023, the IS-IS device can learn only the interface whose route cost exceeds 1023 for the first time. That is, the cost of each interface before this interface is not greater than 63. The routes of the network segment where the interface resides and the routes imported by the interface can all be learned. The cost of the route is 1023. Subsequent routes forwarded by the interface are discarded.
  - If **relax-spf-limit** is specified, the cost of a route works as follows:
    - There is no limit on costs of interfaces or route costs. The cost of a route received by an interface is the actual cost.
  - If the cost style is wide-compatible or wide, the cost of the interface ranges from 1 to 16777215. When the cost is 16777215, the neighbor TLV generated on the link cannot be used for route calculation but for the transmission of TE information. The maximum cost of a received route is 0xFFFFFFFF.
- Configure the cost of an IS-IS interface.
    1. Run:  

```
system-view
```

The system view is displayed.
    2. Run:  

```
interface interface-type interface-number
```

The interface view is displayed.
    3. Run:  

```
isis ipv6 cost cost [level-1 | level-2]
```

The cost of the IS-IS interface is configured.

You can use the **isis ipv6 cost** command to configure the cost of a specified interface.
  - Configure the global IS-IS cost.
    1. Run:  

```
system-view
```

The system view is displayed.
    2. Run:  

```
isis [process-id]
```

The IS-IS view is displayed.
    3. Run:  

```
ipv6 circuit-cost cost [level-1 | level-2]
```

The global IS-IS cost is configured.

You can use the **ipv6 circuit-cost** command to configure the costs of all interfaces at a time.

- Enable IS-IS to automatically calculate interface costs.

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
isis [process-id]
```

The IS-IS view is displayed.

3. Run:

```
ipv6 bandwidth-reference value
```

The reference value of the bandwidth is configured. By default, the bandwidth reference value is 100 Mbit/s.

4. Run:

```
ipv6 auto-cost enable
```

The interface is configured to automatically calculate its cost.

The configuration of the bandwidth reference value takes effect only when the cost type is wide or wide-compatible. In this case, Cost of each interface = (Value of **bandwidth-reference**/Interface bandwidth) x 10.

If the cost-style is narrow, narrow-compatible, or compatible, the cost of each interface is based on costs listed in [Table 7-2](#).

**Table 7-2** Mapping between IS-IS interface costs and interface bandwidth

| Cost | Bandwidth Range                               |
|------|-----------------------------------------------|
| 60   | Interface bandwidth ≤ 10 Mbit/s               |
| 50   | 10 Mbit/s < interface bandwidth ≤ 100 Mbit/s  |
| 40   | 100 Mbit/s < interface bandwidth ≤ 155 Mbit/s |
| 30   | 155 Mbit/s < interface bandwidth ≤ 622 Mbit/s |
| 20   | 622 Mbit/s < Interface bandwidth ≤ 2.5 Gbit/s |
| 10   | Interface bandwidth > 2.5 Gbit/s              |

 **NOTE**

To change the cost of a loopback interface, run the **isis ipv6 cost** command only in the loopback interface view.

----End

## 7.11.5 (Optional) Configuring IPv6 IS-IS Attributes for Interfaces on Different Types of Networks

Different IS-IS attributes can be configured for different types of network interfaces.

### Context

The establishment modes of IS-IS neighbor relationships are different on a broadcast network and on a P2P network. Different IS-IS attributes can be configured for interfaces on different types of networks.

IS-IS is required to select a DIS on a broadcast network. Configure the DIS priorities of IS-IS interfaces so that the interface with the highest priority will be selected as the DIS.

The network types of the IS-IS interfaces on both ends of a link must be the same; otherwise, the IS-IS neighbor relationship cannot be established between the two interfaces. For example, if the type of an interface on a peer device is P2P, you can configure the type of an interface on the local device to P2P so that an IS-IS neighbor relationship can be established between the two devices.

IS-IS on a P2P network is not required to select a DIS. Therefore, you do not need to configure DIS priorities. To ensure the reliability of P2P links, configure IS-IS to use the three-way handshake mode for IS-IS neighbor relationship establishment so that faults on a unidirectional link can be detected.

### Procedure

- Configure the DIS priority of an IS-IS interface.

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
interface interface-type interface-number
```

The interface view is displayed.

3. Run:

```
isis dis-priority priority [level-1 | level-2]
```

The DIS priority is configured on the interface. The greater the value, the higher the priority.

- Configure the network type of an IS-IS interface.

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
interface interface-type interface-number
```

The interface view is displayed.

3. Run:

```
isis circuit-type p2p
```

The network type of the interface is set to P2P.



The network type of an interface is determined by the physical type of the interface by default.

When the network type of an IS-IS interface changes, interface configurations change accordingly.

- After a broadcast interface is configured as a P2P interface using the **isis circuit-type p2p** command, the default settings are restored for the interval for sending Hello packets, the number of Hello packets that IS-IS fails to receive from a neighbor before the neighbor is declared Down, interval for retransmitting LSPs on a P2P link, and various IS-IS authentication modes. Consequently, other configurations such as the DIS priority, DIS name, and interval for sending CSNPs on a broadcast network become invalid.
- After the **undo isis circuit-type** command is run to restore the network type, the default settings are restored for the interval for sending Hello packets, the number of Hello packets that IS-IS fails to receive from a neighbor before the neighbor is declared Down, interval for retransmitting LSPs on a P2P link, various IS-IS authentication modes, DIS priority, and interval for sending CSNPs on a broadcast network.

- Set the negotiation mode in which P2P neighbor relationships can be set up.

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
interface interface-type interface-number
```

The interface view is displayed.

3. Run:

```
isis ppp-negotiation { 2-way | 3-way [only] }
```

The negotiation mode is specified on the interface.

By default, the **3-way** handshake negotiation mode is adopted.

The **isis ppp-negotiation** command can only be used for the establishment of the neighbor relationships on P2P links. In the case of a broadcast link, you can run the **isis circuit-type p2p** command to set the link type to P2P, and then run the **isis ppp-negotiation** command to set the negotiation mode for the establishment of the neighbor relationship.

- Configure OSICP negotiation check on PPP interfaces.

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
interface interface-type interface-number
```

The interface view is displayed.

3. Run:

```
isis ppp-osicp-check
```

The OSICP negotiation status is checked on a PPP interface.

By default, the OSICP negotiation status of a PPP interface does not affect the status of an IS-IS interface.

The **isis ppp-osi-cp-check** command is applicable only to PPP interfaces. This command is invalid for other P2P interfaces.

After this command is run, the OSICP negotiation status of a PPP interface affects the status of an IS-IS interface. When PPP detects that the OSI network fails, the link status of the IS-IS interface goes Down and the route to the network segment where the interface resides is not advertised through LSPs.

- Configure IS-IS not to check whether the IP addresses of received Hello packets are on the same network segment.

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
interface interface-type interface-number
```

The interface view is displayed.

3. Run:

```
isis peer-ip-ignore
```

IS-IS is configured not to check whether the IP addresses of received Hello packets are on the same network segment.

---End

## 7.11.6 Checking the Configuration

After basic IPv6 IS-IS functions are configured, you can view information about IS-IS neighbors, interfaces, and routes.

### Prerequisites

The configurations of basic IPv6 IS-IS functions are complete.

### Procedure

**Step 1** Run the **display isis name-table** [ *process-id* ] command to check the mapping from the name of the local device to the system ID.

**Step 2** Run the **display isis peer** [ *verbose* ] [ *process-id* ] command to check information about IS-IS neighbors.

**Step 3** Run the **display isis interface** [ *verbose* ] [ *process-id* ] command to check information about IS-IS interfaces.

**Step 4** Run the **display isis route** [ *process-id* | **vpn-instance** *vpn-instance-name* ] **ipv6** [ *verbose* ] [ *level-1* | *level-2* ] [ *ipv6-address* [ *prefix-length* ] ] \* [ *count* ] command to check information about IS-IS routes.

---End

## 7.12 Configuring IPv6 IS-IS Route Selection

Configuring IS-IS route selection can achieve refined control over route selection.

### 7.12.1 Establishing the Configuration Task

Before configuring IPv6 IS-IS route selection, familiarize yourself with the applicable environment, complete the pre-configuration tasks, and obtain the data required for the configuration. This will help you complete the configuration task quickly and efficiently.

#### Applicable Environment

After basic IPv6 IS-IS functions are configured, IS-IS routes will be generated, enabling communication between different nodes on a network.

If multiple routes are available, a route discovered by IS-IS may not be the optimal route. This does not meet network planning requirements nor facilitates traffic management. Therefore, configure IPv6 IS-IS route selection to implement refined control over route selection.

To implement refined control over IPv6 IS-IS route selection, perform the following operations:

- **Configuring the IPv6 IS-IS Interfaces.**

 **NOTE**

Changing the IS-IS cost for an interface can achieve the function of controlling route selection, but requires routes on the interface to be recalculated and reconverged when a network topology changes, especially on a large-scale network. In addition, the configuration result may not meet your expectation.

Therefore, the configuration of changing IS-IS costs has best to be finished when configuring basic IS-IS functions.

- Configure IPv6 IS-IS route leaking.
- Configure principles for using equal-cost IPv6 IS-IS routes.
- Filter IPv6 IS-IS routes.
- Configure an overload bit for an IPv6 IS-IS device.

#### Pre-configuration Tasks

Before configuring IPv6 IS-IS route selection, complete the following tasks:

- Configuring the link layer protocol on interfaces.
- Configuring IP addresses for interfaces to ensure that neighboring nodes are reachable at the network layer.
- **Configuring Basic IPv6 IS-IS Functions.**

#### Data Preparation

To configure the IPv6 IS-IS route selection, you need the following data.

| No. | Data                                                           |
|-----|----------------------------------------------------------------|
| 1   | ACL6 for filtering routes, IPv6 prefix list, or routing policy |

| No. | Data                                                     |
|-----|----------------------------------------------------------|
| 2   | Maximum number of load-balancing equal-cost IS-IS routes |
| 3   | Time when an IS-IS device enters the overload state      |

## 7.12.2 Configuring IPv6 IS-IS Route Leaking

Configuring IS-IS route leaking enables you to optimize IS-IS route selection on a two-level-area network.

### Context

If multiple Level-1-2 devices in a Level-1 area are connected to devices in the Level-2 area, a Level-1 LSP sent by each Level-1-2 device carries an ATT flag bit of 1. This Level-1 area will have multiple routes to the Level-2 area and to other Level-1 areas.

By default, routes in a Level-1 area can be leaked into the Level-2 area so that Level-1-2 and Level-2 devices can learn about the topology of the entire network. Devices in a Level-1 area are unaware of the entire network topology because they only maintain LSDBs in the local Level-1 area. Therefore, a device in a Level-1 area can forward traffic to a Level-2 device only through the nearest Level-1-2 device. The route used may not be the optimal route to the destination.

To enable a device in a Level-1 area to select the optimal route, configure IPv6 IS-IS route leaking so that specified routes in the Level-2 area can be leaked into the local Level-1 area.

Routes of services deployed only in the local Level-1 area do not need to be leaked into the Level-2 area. A policy can be configured to leak only desired routes into the Level-2 area.

### Procedure

- Configure routes in the Level-2 area to leak into Level-1 area.

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
isis [process-id]
```

The IS-IS view is displayed.

3. Run:

```
ipv6 import-route isis level-2 into level-1 [tag tag | filter-policy
{ ipv6-prefix ipv6-prefix-name | route-policy route-policy-name }] *
```

Routes in the Level-2 area and other Level-1 areas that meet the specified conditions are leaked into the local Level-1 area.

#### NOTE

The command is run on the Level-1-2 device that is connected to an external area.

By default, routes in the Level-2 area are not leaked into Level-1 areas. After this command is run, only routes that meet the specified conditions can be leaked into Level-1 areas.

- Configure routes in Level-1 areas to leak into the Level-2 area.

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
isis [process-id]
```

The IS-IS view is displayed.

3. Run:

```
ipv6 import-route isis level-1 into level-2 [tag tag | filter-policy
{ ipv6-prefix ipv6-prefix-name | route-policy route-policy-name }] *
```

Routes that meet the specifies conditions in Level-1 areas are leaked into the Level-2 area.

 **NOTE**

The command is run on the Level-1-2 device that is connected to an external area.

By default, all routes in a Level-1 area are leaked into the Level-2 area. After this command is run, only routes that meet the specified conditions can be leaked into the Level-2 area.

----End

## 7.12.3 Configuring Principles for Using Equal-Cost IPv6 IS-IS Routes

If multiple equal-cost IS-IS routes are available on a network, configure the equal-cost IS-IS routes to work in load-balancing mode to increase the bandwidth usage of each link.

### Context

If there are redundant IPv6 IS-IS links, multiple routes may have an equal cost. Configure the equal-cost IPv6 IS-IS routes to work in load-balancing mode so that traffic will be evenly balanced among these links.

This mechanism increases the link bandwidth usage and prevents network congestion caused by link overload. However, this mechanism may make traffic management more difficult because traffic will be randomly forwarded.

### Procedure

- Step 1** Run:

```
system-view
```

The system view is displayed.

- Step 2** Run:

```
isis [process-id]
```

The IS-IS view is displayed.

- Step 3** Run:

```
ipv6 maximum load-balancing number
```

The maximum number of load-balancing equal-cost IS-IS routes is set.

 **NOTE**

If the number of IS-IS equal-cost routes is greater than the value of *number*, the number of IS-IS equal-cost routes to work in load-balancing mode is determined by *number*. If the number of IS-IS equal-cost routes is smaller than the value of *number*, IS-IS equal-cost routes of the actual number work in load-balancing mode.

----End

## 7.12.4 Filtering IPv6 IS-IS Routes

If some IS-IS routes are not preferred, configure conditions to filter IS-IS routes. Only IS-IS routes meeting the specified conditions can be added to an IP routing table.

### Context

Only routes in an IP routing table can be used to forward IP packets. An IS-IS route can take effect only after this IS-IS route has been successfully added to an IP routing table.

If an IS-IS route does not need to be added to a routing table, specify conditions, such as IPv6 prefix, and routing policy, to filter routes so that only IS-IS routes that meet the specified conditions can be added to an IP routing table. IS-IS routes that do not meet the specified conditions cannot be added to the IP routing table and cannot be selected to forward IP packets.

### Procedure

**Step 1** Run:

```
system-view
```

The system view is displayed.

**Step 2** Run:

```
isis [process-id]
```

The IS-IS view is displayed.

**Step 3** Run:

```
ipv6 filter-policy { ipv6-prefix ipv6-prefix-name | route-policy route-policy-name } import
```

Conditions for filtering IS-IS routes are configured.

----End

## 7.12.5 Configuring an Overload Bit for an IPv6 IS-IS Device

If an IS-IS device needs to be temporarily isolated, configure the IS-IS device to enter the overload state to prevent other devices from forwarding traffic to this IS-IS device and prevent blackhole routes.

### Context

If an IS (for example, an IS to be upgraded or maintained) needs to be temporarily isolated, configure the IS to enter the overload state so that no device will forward traffic to this IS.

IS-IS routes converge more quickly than BGP routes. To prevent blackhole routes on a network where both IS-IS and BGP are configured, set an overload bit to instruct an IS to enter the

overload state during its start or restart. After BGP convergence is complete, cancel the overload bit.

## Procedure

**Step 1** Run:

```
system-view
```

The system view is displayed.

**Step 2** Run:

```
isis [process-id]
```

The IS-IS view is displayed.

**Step 3** Run:

```
set-overload [on-startup [timeout1 | start-from-nbr system-id [timeout1
[timeout2]] | wait-for-bgp [timeout1]]] [allow { interlevel | external }
*]
```

The overload bit is configured.

----End

## 7.12.6 Checking the Configuration

After configuring IPv6 IS-IS route selection, run the following commands to verify that the configurations are correct.

### Procedure

- Run the **display isis route** [ *process-id* | **vpn-instance** *vpn-instance-name* ] [ **ipv6** ] [ **verbose** | [ **level-1** | **level-2** ] | *ipv6-address* [ *prefix-length* ] ] \* [ **count** ] command to check IS-IS routing information.
- Run the **display isis lsdb** [ { **level-1** | **level-2** } | **verbose** | { **local** | *lsp-id* | **is-name** *symbolic-name* } ] \* [ *process-id* | **vpn-instance** *vpn-instance-name* ] command to check information in the IS-IS LSDB.

----End

## 7.13 Configuring IPv6 IS-IS Route Summarization

To improve the route searching efficiency and simplify route management on a large-scale IS-IS network, configure IS-IS route summarization to reduce the number of IS-IS routes in a routing table.

### Context

Route summarization is used to summarize routes with the same IP prefix into one route.

On a large-scale IS-IS network, route summarization can be configured to reduce the number of IS-IS routes in a routing table. This summarization improves the usage of system resources and facilitates route management.

If a link on an IP network segment that is summarized frequently alternates between Up and Down states, IP network segments that are not summarized will not be affected, preventing route flapping and improving the network stability.

## Procedure

### Step 1 Run:

```
system-view
```

The system view is displayed.

### Step 2 Run:

```
isis [process-id]
```

The IS-IS view is displayed.

### Step 3 Run:

```
ipv6 summary ipv6-address prefix-length [avoid-feedback | generate_null0_route |
tag tag | [level-1 | level-1-2 | level-2]] *
```

The specified IS-IS routes are summarized into one IS-IS route.

#### NOTE

After route summarization is configured on an IS, the local routing table still contains all specific routes before the summarization.

The routing tables on other ISs contain only the summary route, and the summary route is deleted only after all its specific routes are deleted.

----End

## Checking the Configuration

After the route summarization function is configured, perform the following steps to check whether the route summarization function has taken effect.

- Run the **display isis route** command to check summary routes in the IS-IS routing table.
- Run the **display ipv6 routing-table [ verbose ]** command to check summary routes in the IP routing table.

## 7.14 Configuring IPv6 IS-IS to Interact with Other Routing Protocols

If other routing protocols are configured on an IS-IS network, you need to configure IS-IS to interact with these protocols to ensure successful communication between them.

### 7.14.1 Establishing the Configuration Task

Before configuring IPv6 IS-IS to interact with other routing protocols, familiarize yourself with the applicable environment, complete the pre-configuration tasks, and obtain the data required for the configuration. This will help you complete the configuration task quickly and efficiently.

#### Applicable Environment

If other routing protocols are configured on an IS-IS network, the following issues need to be considered:



- Preference of IS-IS routes

If multiple routes to the same destination are discovered by different routing protocols running on the same device, the route discovered by the protocol with the highest preference is selected. For example, if both OSPFv3 and IS-IS are configured, the route discovered by OSPFv3 is used because OSPFv3 enjoys a higher preference than IS-IS by default.

Therefore, if you want the route discovered by IS-IS to be used, configure IS-IS to have the highest preference.

- Communication between an IS-IS area and other areas

If other routing protocols are configured on an IS-IS network, you need to configure IS-IS to interact with those routing protocols so that IS-IS areas can communicate with non-IS-IS areas.

 **NOTE**

The LSDBs of different IS-IS processes on a device are independent of each other. Therefore, each IS-IS process on the device considers routes of the other IS-IS processes as external routes.

To ensure successful traffic forwarding, configure IS-IS to interact with other routing protocols on a device where external routes are configured, for example, a Level-1-2 IS-IS router. Available methods are as follows:

- Configure IS-IS to advertise a default route.

This mode is easy to configure and does not require devices in IS-IS areas to learn external routes. After a default route is advertised, all traffic in an IS-IS area is forwarded through the default route.

- Configure IS-IS to import external routes.

This mode enables all devices in IS-IS areas to learn external routes, implementing refined control over traffic forwarding.

To ensure successful forwarding of traffic destined for IS-IS areas, you must also enable the other routing protocols to interact with IS-IS.

## Pre-configuration Tasks

Before configuring IPv6 IS-IS to interact with other routing protocols, complete the following tasks:

- Configuring the link layer protocol on interfaces
- Configuring IP addresses for interfaces to ensure that neighboring nodes are reachable at the network layer
- **Configuring Basic IPv6 IS-IS Functions**
- Configuring basic functions of other routing protocols

## Data Preparation

To configure the IPv6 IS-IS to interact with other routing protocols, you need the following data.

| No. | Data                                |
|-----|-------------------------------------|
| 1   | IPv6 prefix list, or routing policy |
| 2   | Preference value of IS-IS           |

## 7.14.2 Configuring a Preference Value for IPv6 IS-IS

If multiple routes to the same destination are discovered by different routing protocols, configuring the highest preference value for IS-IS allows a route discovered by IS-IS to be selected preferentially.

### Context

If multiple routes to the same destination are discovered by different routing protocols running on the same device, the route discovered by the protocol with the highest preference is selected.

For example, if both OSPFv3 and IS-IS are configured on a network, the route discovered by OSPFv3 is used because OSPFv3 has a higher preference than IS-IS by default.

To prefer a route discovered by IS-IS, configure a higher preference value for IS-IS. In addition, a routing policy can be configured to increase the preferences of specified IS-IS routes, without affecting route selection.

### Procedure

- Configure the IS-IS preference value.

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
isis [process-id]
```

The IS-IS view is displayed.

3. Run:

```
ipv6 preference preference
```

The IS-IS preference value is configured.

#### NOTE

A smaller *preference* value indicates a higher preference.

The default IS-IS preference value is **15**.

- Configure preference values for specified IS-IS routes.

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
isis [process-id]
```

The IS-IS view is displayed.

3. Run:

```
ipv6 preference route-policy route-policy-name preference
```

The preference values are configured for the specified IS-IS routes.

#### NOTE

*preference* takes effect only for IS-IS routes that match the specified routing policy.

----End

## 7.14.3 Configuring IPv6 IS-IS to Advertise a Default Route

To forward all traffic in an IS-IS area through a default route, configure IS-IS on a Level-1-2 device to advertise the default route.

### Context

Only the route `::/0` can be advertised as a default route on a Level-1-2 device. All traffic destined for other areas is first forwarded to the Level-1-2 device.

To ensure successful traffic forwarding, external routes must be learned on the Level-1-2 device.

#### NOTE

Configuring static default routes can also achieve the function of interaction between different routing protocols, but require large configurations and are difficult to manage.

If multiple Level-1-2 devices are deployed, a routing policy can be configured to allow only the Level-1-2 device that meets the specified conditions to advertise a default route, preventing blackhole routes.

### Procedure

#### Step 1 Run:

```
system-view
```

The system view is displayed.

#### Step 2 Run:

```
isis [process-id]
```

The IS-IS view is displayed.

#### Step 3 Run:

```
ipv6 default-route-advertise [always | match default | route-policy route-policy-name] [cost cost | tag tag | [level-1 | level-1-2 | level-2]] * [avoid-learning]
```

IS-IS is configured to advertise a default route.

----End

## 7.14.4 Configuring IPv6 IS-IS to Import External Routes

If devices in an IS-IS area need to learn external routes, configure IS-IS on a Level-1-2 device of this area to import external routes.

### Context

If IS-IS is configured on a Level-1-2 device to advertise a default route, all traffic in IS-IS areas will be forwarded by this Level-1-2 device. This will burden this Level-1-2 device because no external route can be learned on the devices in the IS-IS areas.

If multiple Level-1-2 devices are deployed, optimal routes to other areas need to be selected. To ensure optimal routes are selected, all the other devices in the IS-IS areas must learn all or some external routes.

Routing policies can be configured to import or advertise external routes that meet specified conditions to the IS-IS areas.

## Procedure

- Configure IS-IS to import external routes.

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
isis [process-id]
```

The IS-IS view is displayed.

3. Run:

```
ipv6 import-route
```

IS-IS is configured to import external routes.

### NOTE

IS-IS will advertise all imported external routes to the IS-IS areas by default.

If only some imported external routes need to be advertised, run the **ipv6 filter-policy export** command to set a filtering policy.

- (Optional) Configure IS-IS to advertise some external routes to the IS-IS areas.

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
isis [process-id]
```

The IS-IS view is displayed.

3. Run:

```
ipv6 filter-policy { ipv6-prefix ipv6-prefix-name | route-policy route-policy-name } export [protocol [process-id]]
```

IS-IS is configured to advertise specified external routes to the IS-IS areas.

### NOTE

After this command is run, only external routes that meet the specified conditions can be advertised to the IS-IS areas.

----End

## 7.14.5 Checking the Configuration

After IS-IS is enabled to import routes from other protocols, run the following commands to verify that the configurations are correct.

## Procedure

- Run the **display isis lsdb** [ { level-1 | level-2 } | verbose | { local | lsp-id | is-name symbolic-name } ] \* [ process-id | vpn-instance vpn-instance-name ] command to check IS-IS LSDB information.
- Run the **display isis route** [ process-id | vpn-instance vpn-instance-name ] [ ipv6 ] [ verbose | [ level-1 | level-2 ] | ipv6-address [ prefix-length ] ] \* [ | count ] command to check IS-IS routing information.

- Run the **display ipv6 routing-table ipv6-prefix *ipv6-prefix-name* [ verbose ]** command to check the IP routing table.

---End

## 7.15 Configuring the IPv6 IS-IS Route Convergence Speed

Accelerating IS-IS route convergence can improve the fault location efficiency and improve the network reliability.

### 7.15.1 Establishing the Configuration Task

Before configuring the IPv6 IS-IS route convergence speed, familiarize yourself with the applicable environment, complete the pre-configuration tasks, and obtain the data required for the configuration. This will help you complete the configuration task quickly and efficiently.

#### Applicable Environment

The procedure for implementing IS-IS is as follows:

- Establishment of neighboring relationships: establishes neighboring relationships by exchanging Hello packets between two devices.
- LSP flooding: implements LSDB synchronization between devices in the same area.
- SPF calculation: uses the SPF algorithm to calculate IS-IS routes, and delivers the IS-IS routes to the routing table.

To accelerate the IS-IS route convergence speed, configure the following parameters:

- Interval for detecting IS-IS neighboring device failures.
- Flooding parameters of CSNPs and LSPs.
- Interval for SPF calculation.

You can also configure convergence priorities for IPv6 IS-IS routes so that key routes can be converged by preference when a network topology changes. This minimizes adverse impacts on key services.

#### Pre-configuration Tasks

Before configuring the IPv6 IS-IS route convergence speed, complete the following tasks:

- Configuring the link layer protocol on interfaces.
- Configuring IP addresses for interfaces to ensure that neighboring nodes are reachable at the network layer.
- [Configuring Basic IPv6 IS-IS Functions](#).

#### Data Preparation

To configure the IPv6 IS-IS route convergence speed, you need the following data.

| No. | Data                                                                                 |
|-----|--------------------------------------------------------------------------------------|
| 1   | Interval at which Hello packets are sent and the holding time of neighboring devices |

| No. | Data                            |
|-----|---------------------------------|
| 2   | Flooding time of CSNPs and LSPs |
| 3   | Interval for SPF calculation    |
| 4   | Route convergence priority      |

## 7.15.2 Configuring the Interval for Detecting IS-IS Neighboring Device Failures

To minimize the effects caused by neighboring device failures on an IS-IS network, accelerate the speed of detecting IS-IS neighboring device failures.

### Context

Connection status between an IS-IS device and its neighboring devices can be monitored by exchanging Hello packets at intervals. An IS-IS neighboring device is considered Down if the IS-IS device does not receive any Hello packets from the neighboring device within the specified period (called the holding time). A failure in an IS-IS neighboring device will trigger LSP flooding and SPF calculation, after which IS-IS routes are reconverged.

To speed up fault detection, use the following methods to accelerate the speed of detecting IS-IS neighboring device failures:

- Set an interval at which Hello packets are sent.
- Set the holding multiplier for neighboring devices.

### Procedure

- Set an interval at which Hello packets are sent.
  1. Run:  

```
system-view
```

The system view is displayed.
  2. Run:  

```
interface interface-type interface-number
```

The interface view is displayed.
  3. Run:  

```
isis timer hello hello-interval [level-1 | level-2]
```

The interval at which Hello packets are sent is set.

#### NOTE

A broadcast link can transmit both Level-1 and Level-2 Hello packets. You can set different sending intervals for these two types of Hello packets. By default, both Level-1 and Level-2 Hello packets are sent.

A P2P link can transmit only one type of Hello packets. Therefore, there is no need to specify the **level-1** or **level-2** parameter if a P2P link is used.

- Set the holding multiplier for neighboring devices.

1. Run:  
`system-view`  
The system view is displayed.
2. Run:  
`interface interface-type interface-number`  
The interface view is displayed.
3. Run:  
`isis timer holding-multiplier number [ level-1 | level-2 ]`  
The holding multiplier of neighboring devices is set.

---End

### 7.15.3 Setting Flooding Parameters of SNPs and LSPs

To speed up LSDB synchronization between devices, set flooding parameters of SNPs and LSPs to proper values.

#### Context

SNPs consist of CSNPs and PSNPs. CSNPs carry summaries of all LSPs in LSDBs, ensuring LSDB synchronization between neighboring routers. SNPs are processed differently on broadcast links and P2P links.

- On a broadcast link, CSNPs are periodically sent by a DIS device. If a router detects that its LSDB is not synchronized with that on its neighboring router, the router will send PSNPs to apply for missing LSPs.
- On a P2P link, CSNPs are sent only during initial establishment of neighboring relationships. If a request is acknowledged, a neighboring router will send a PSNP in response to a CSNP. If a router detects that its LSDB is not synchronized with that on its neighboring router, the router will also send PSNPs to apply for missing LSPs.

To speed up LSDB synchronization, modify the following parameters of SNPs and LSPs on the AR150/200:

- **Set an interval at which CSNPs are sent.**
- **Configure the intelligent timer controlling LSP generation.**
- **Set the maximum length for LSPs.**
- **Set the refresh interval for LSPs.**
- **Set the maximum lifetime for LSPs.**
- **Set the minimum interval at which LSPs are sent.**
- **Enable LSP fast flooding.**
- **Set an interval at which LSPs are retransmitted over a P2P link.**

#### Procedure

- Set an interval at which CSNPs are sent.
  1. Run:  
`system-view`  
The system view is displayed.

2. Run:  
`interface interface-type interface-number`

The interface view is displayed.

3. Run:  
`isis timer csnp csnp-interval [ level-1 | level-2 ]`

The interval at which CSNPs are sent is set on the specified interface.

 **NOTE**

Configure **Level-1** and **Level-2** only when a broadcast interface is specified.

- Configure the intelligent timer controlling LSP generation.

1. Run:  
`system-view`

The system view is displayed.

2. Run:  
`isis [ process-id ]`

The IS-IS view is displayed.

3. Run:  
`timer lsp-generation max-interval [ init-interval [ incr-interval ] ]  
[ level-1 | level-2 ]`

The intelligent timer controlling LSP generation is configured.

If a level is not specified, both **level-1** and **level-2** are used by default.

The delay in generating an LSP or an LSP fragment for the first time is determined by *init-interval*; the delay in generating an LSP or an LSP fragment for the second time is determined by *incr-interval*. From the third time on, the delay in generating an LSP increases twice every time until the delay reaches the value specified by *max-interval*. After the delay remains at the value specified by *max-interval* for three times or the IS-IS process is restarted, the delay decreases to the value specified by *init-interval*.

If *incr-interval* is not specified, the delay in generating an LSP or LSP fragment for the first time is determined by *init-interval*. From the second time on, the delay in generating an LSP is determined by *max-interval*. After the delay remains at the value specified by *max-interval* for three times or the IS-IS process is restarted, the delay decreases to the value specified by *init-interval*.

When only *max-interval* is specified, the intelligent timer functions as an ordinary one-time triggering timer.

- Set the maximum length for LSPs.

1. Run:  
`system-view`

The system view is displayed.

2. Run:  
`isis [ process-id ]`

The IS-IS view is displayed.

3. Run:  
`lsp-length originate max-size`



The maximum length is set for each LSP to be generated.

4. Run:

```
lsp-length receive max-size
```

The maximum length is set for each LSP to be received.

 **NOTE**

Ensure that the value of *max-size* for LSPs to be generated must be smaller than or equal to the value of *max-size* for LSPs to be received.

The value of *max-size* in the **lsp-length** command must meet the following conditions.

- The MTU of an Ethernet interface must be greater than or equal to the sum of the value of *max-size* and 3.
- The MTU of a P2P interface must be greater than or equal to the value of *max-size*.

● Set the refresh interval for LSPs.

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
isis [process-id]
```

The IS-IS view is displayed.

3. Run:

```
timer lsp-refresh refresh-time
```

A refresh interval is set for LSPs.

To synchronize all LSPs in the areas, IS-IS regularly transmits all the current LSPs to neighbors.

By default, the LSP refresh interval is 900s, and the maximum lifetime of an LSP is 1200s. Ensure that the LSP refresh interval is more than 300s shorter than the maximum LSP lifetime. This allows new LSPs to reach all routers in an area before existing LSPs expire.

 **NOTE**

The larger a network, the greater the deviation between the LSP refresh interval and the maximum LSP lifetime.

● Set the maximum lifetime for LSPs.

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
isis [process-id]
```

The IS-IS view is displayed.

3. Run:

```
timer lsp-max-age age-time
```

The maximum lifetime is set for LSPs.

When a router generates the system LSP, it fills in the maximum lifetime for this LSP. After this LSP is received by other routers, the lifetime of the LSP is reduced gradually. If the router does not receive any more update LSPs and the lifetime of the LSP is reduced to 0, the LSP will be deleted from the LSDB 60s later if no more updated LSPs are received.

- Set the minimum interval at which LSPs are sent.

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
interface interface-type interface-number
```

The interface view is displayed.

3. Run:

```
isis timer lsp-throttle throttle-interval [count count]
```

The minimum interval at which LSPs are sent is set.

The *count* parameter specifies the maximum number of LSPs that can be sent within the interval specified by *throttle-interval*. The value of *count* is an integer ranging from 1 to 1000.

- Enable LSP fast flooding.

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
isis [process-id]
```

The IS-IS view is displayed.

3. Run:

```
flash-flood [lsp-count | max-timer-interval interval | [level-1 | level-2]] *
```

The LSP fast flooding is enabled.

Running the **flash-flood** command speeds up LSP flooding. The *lsp-count* parameter specifies the number of LSPs flooded each time, which is applicable to all interfaces. If the number of LSPs to be sent is greater than the value of *lsp-count*, *lsp-count* takes effect. If the number of LSPs to be sent is smaller than the value of *lsp-count*, LSPs of the actual number are sent. If a timer is configured and the configured timer does not expire before the route calculation, the LSPs are flooded immediately when being received; otherwise, the LSPs are sent when the timer expires.

When LSP fast flooding is enabled, Level-1 LSPs and Level-2 LSPs are fast flooded by default if no level is specified.

- Set an interval at which LSPs are retransmitted over a P2P link.

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
interface interface-type interface-number
```

The interface view is displayed.

3. Run:

```
isis timer lsp-retransmit retransmit-interval
```

The interval at which LSPs are retransmitted over a P2P link is set.

----End

## 7.15.4 Setting the SPF Calculation Interval

To improve the fault location efficiency on an IS-IS network and prevent SPF calculation from consuming excessive system resources, set the SPF calculation interval to a proper value.

### Context

A network change always triggers IS-IS to perform SPF calculation. Frequent SPF calculation will consume excessive CPU resources, affecting services.

To solve this problem, configure an intelligent timer to control the interval for SPF calculation. For example, to speed up IS-IS route convergence, set the interval for SPF calculation to a small value, and set the interval to a large value after the IS-IS network becomes stable.

### Procedure

**Step 1** Run:

```
system-view
```

The system view is displayed.

**Step 2** Run:

```
isis [process-id]
```

The IS-IS view is displayed.

**Step 3** Run:

```
timer spf max-interval [init-interval [incr-interval]]
```

The SPF intelligent timer is configured.

The intelligent timer changes as follows:

- The delay for the first SPF calculation is determined by *init-interval*; the delay for the second SPF calculation is determined by *incr-interval*. From the third time on, the delay in SPF calculation increases twice every time until the delay reaches the value specified by *max-interval*. After the delay remains at the value specified by *max-interval* for three times or the IS-IS process is restarted, the delay decreases to the value specified by *init-interval*.
- If *incr-interval* is not specified, the delay in SPF calculation for the first time is determined by *init-interval*. From the second time on, the delay in SPF calculation is determined by *max-interval*. After the delay remains at the value specified by *max-interval* for three times or the IS-IS process is restarted, the delay decreases to the value specified by *init-interval*.
- When only *max-interval* is specified, the intelligent timer functions as an ordinary one-time triggering timer.

----End

## 7.15.5 Configuring Convergence Priorities for IPv6 IS-IS Routes

If some IS-IS routes need to be converged by preference to minimize adverse impacts on services, configure those routes to have the highest convergence priority.

### Context

By default, the convergence priority of 128-bit host routes is **medium**, and the convergence priority of the other IS-IS routes is **low**.

The AR150/200 allows you to configure the highest convergence priority for specific IS-IS routes so that those IS-IS routes will be converged first when a network topology changes.

### Procedure

**Step 1** Run:

```
system-view
```

The system view is displayed.

**Step 2** Run:

```
isis [process-id]
```

The IS-IS view is displayed.

**Step 3** Run:

```
ipv6 prefix-priority [level-1 | level-2] { critical | high | medium } { ipv6-
prefix prefix-name | tag tag-value }
```

Convergence priorities are set for IS-IS routes.

The application rules of the convergence priorities for IS-IS routes are as follows:

- Existing IS-IS routes are converged based on the priorities configured in the **ipv6 prefix-priority** command.
- New IS-IS routes are converged based on the priorities configured in the **ipv6 prefix-priority** command.
- If an IS-IS route conforms to the matching rules of multiple convergence priorities, the highest convergence priority is used.
- The convergence priority of a Level-1 IS-IS route is higher than that of a Level-2 IS-IS route.
- If the route level is not specified, the configuration of the **prefix-priority** command takes effect for both Level-1 and Level-2 IS-IS routes.

 **NOTE**

The **ipv6 prefix-priority** command is only applicable to the public network.

After the **ipv6 prefix-priority** command is run, the convergence priority of 32-bit host routes is **low**, and the convergence priorities of the other routes are determined as specified in the **ipv6 prefix-priority** command.

----End

## 7.15.6 Checking the Configuration

After the parameters specifying the IPv6 IS-IS route convergence speed are set, run the following commands to verify that the configurations are correct.

## Procedure

- Run the **display isis interface** [ **verbose** ] [ *process-id* | **vpn-instance** *vpn-instance-name* ] command to check IS-IS packet information.
- Run the **display isis route** [ *process-id* | **vpn-instance** *vpn-instance-name* ] **ipv6** [ **verbose** | [ **level-1** | **level-2** ] | *ipv6-address* [ *prefix-length* ] ] \* [ | **count** ] command to check the preference of IS-IS routes.

----End

## 7.16 Configuring IS-IS GR

By configuring IS-IS GR, you can enable Router to restart gracefully and avoid temporary black holes.

### 7.16.1 Establishing the Configuration Task

Before configuring IS-IS GR, familiarize yourself with the applicable environment, complete the pre-configuration tasks, and obtain the required data. This can help you complete the configuration task quickly and accurately.

#### Applicable Environment

The restart of an IS-IS router causes the temporary interruption of the network, because the adjacency relationship between the router and its neighbor is torn down. The LSPs packets of the router are deleted, which makes route calculation inaccurate. Packets are thus lost.

You can configure IS-IS GR to solve this problem. After IS-IS GR is enabled, the router notifies the neighbor of the restart status, and reestablishes the adjacency relationship with its neighbor without interrupting the forwarding.

The advantages of IS-IS GR are as follows:

- When IS-IS restarts, the router can resend connection requests to its neighbor. The adjacency relationship is not torn down.
- Before LSPs packets are generated, GR minimizes the interference caused by waiting for the database synchronization.
- If the router starts for the first time, the router sets the overload bit in LSPs until the LSDB synchronization is complete. This avoids route black holes.

#### NOTE

The AR150/200 can function as only the Helper router, but cannot function as the Restarter router.

#### Pre-configuration Tasks

Before configuring IS-IS GR, complete the following tasks:

- Configuring IP addresses for interfaces to ensure network connectivity between neighboring nodes.
- [Configuring Basic IPv4 IS-IS Functions](#)

#### Data Preparation

To configure IS-IS GR, you need the following data.

| No. | Data                                                                                  |
|-----|---------------------------------------------------------------------------------------|
| 1   | ID of an IS-IS process                                                                |
| 2   | Interval for reestablishing GR sessions                                               |
| 3   | Whether to suppress the advertisement of the adjacency when the GR restarter restarts |

## 7.16.2 Enabling IS-IS GR

Before configuring IS-IS GR, you need to enable the GR capability for IS-IS.

### Context

Do as follows on the router that runs IS-IS.

### Procedure

- Step 1** Run:  
`system-view`  
The system view is displayed.
- Step 2** Run:  
`isis [ process-id ]`  
The IS-IS view is displayed.
- Step 3** Run:  
`graceful-restart`  
IS-IS GR is enabled.  
By default, IS-IS GR is disabled.  
---End

## 7.16.3 Configuring Parameters of an IS-IS GR Session

By setting IS-IS GR parameters, you can avoid temporary black holes on the network.

### Context

The router that starts for the first time does not maintain the forwarding status. If the router restarts, the LSPs generated when the router runs last time may exist in the LSDB of other routers in the network.

The sequence number of an LSP fragment is reinitialized when the router starts. Therefore, the router considers that the previously advertised LSP stored on other routers is newer than the LSP generated locally after the router starts. This leads to the temporary black hole in the network, which lasts until the normal LSDB update process finishes. The router then regenerates its LSPs and advertises the LSPs with the highest sequence number.

When this router starts, if the neighbor of the router suppresses the advertisement of the adjacency until this router advertises the updated LSPs, the preceding case can thus be avoided.

Do as follows on the router that runs IS-IS:

## Procedure

**Step 1** Run:

```
system-view
```

The system view is displayed.

**Step 2** Run:

```
isis [process-id]
```

The IS-IS view is displayed.

**Step 3** Run:

```
graceful-restart interval interval-value
```

The interval for reestablishing an IS-IS GR session is set.

The restart interval is set to the Holdtime in an IS-IS Hello PDU. Thus, the adjacency relationship is not torn down when the router restarts. By default, the restart period is 300 seconds.

**Step 4** (Optional) Run:

```
graceful-restart suppress-sa
```

The GR restarter is configured to suppress the Suppress-Advertisement (SA) bit of the restart TLV.

To prevent a router from suppressing the SA bit in a Hello PDU during the active/standby switchover, the administrator can run the **undo graceful-restart suppress-sa** command.

By default, the SA bit is not suppressed.

----End

## 7.16.4 Checking the Configuration

After configuring IS-IS GR, you can check the IS-IS GR status and parameters.

### Prerequisites

The configurations for IS-IS GR are complete.

### Procedure

**Step 1** Run **display isis graceful-restart status [ level-1 | level-2 ] [ process-id | vpn-instance vpn-instance-name ]** command to check the status of IS-IS GR.

----End

## 7.17 Maintaining IS-IS

Maintaining IS-IS involves resetting IS-IS and clearing IS-IS statistics.

## 7.17.1 Resetting IS-IS Data Structure

By restarting IS-IS, you can reset IS-IS. You can also reset IS-IS in GR mode.

### Context



#### CAUTION

The IS-IS data structure cannot be restored after you reset it. All the previous structure information and the neighbor relationship are reset. Exercise caution when running this command.

---

To clear the IS-IS data structure, run the following **reset** command in the user view.

### Procedure

**Step 1** Run **reset isis all** [ [ *process-id* | **vpn-instance** *vpn-instance-name* ] | **graceful-restart** ] \* command to reset the IS-IS data structure.

By default, the IS-IS data structure is not reset.

----End

## 7.17.2 Resetting a Specific IS-IS Neighbor

By restarting IS-IS neighbors, you can reset the IS-IS neighbor relationship, and thus make the new configuration take effect.

### Context



#### CAUTION

The specified IS-IS neighbor relationship is deleted after you reset a specified IS-IS neighbor by using the **reset isis peer** command. Exercise caution when running this command.

---

After the IS-IS routing policy or the protocol changes, you can reset a specific IS-IS neighbor to validate the new configuration.

To reset a specific IS-IS neighbor, run the following **reset** command in the user view.

### Procedure

**Step 1** Run **reset isis peer** *system-id* [ *process-id* | **vpn-instance** *vpn-instance-name* ] command to reset a specific IS-IS neighbor.

----End



## 7.18 Configuration Examples

This section provides several configuration examples of IS-IS together with the configuration flowchart. The configuration examples explain networking requirements, configuration notes, and configuration roadmap.

### 7.18.1 Example for Configuring Basic IS-IS Functions

This part provides an example for interconnecting IPv4 networks through IS-IS.

#### Networking Requirements

As shown in [Figure 7-3](#):

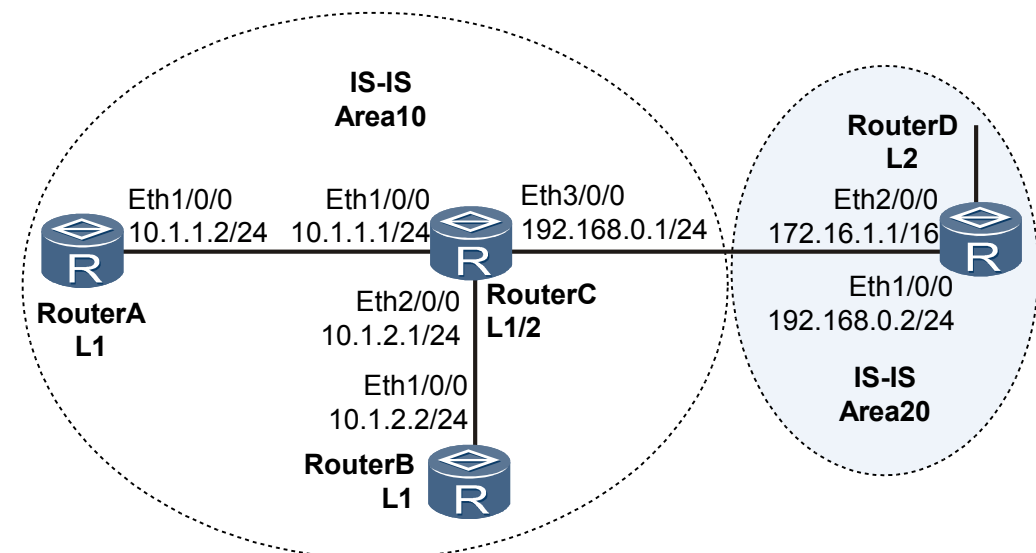
- Router A, Router B, Router C, and Router D belong to the same AS. IS-IS is enabled on the routers to implement interconnection in the IP network.
- The area addresses of Router A, Router B, and Router C are all 10, and the area address of Router D is 20.
- Router A and Router B are Level-1 routers, Router C is a Level-1-2 router. Router D is a Level-2 router.



**NOTE**

AR150/200 is RouterA, or RouterB.

**Figure 7-3** Networking diagram for configuring basic IS-IS functions



#### Configuration Roadmap

The configuration roadmap is as follows:

1. Enable IS-IS on each router, configure the levels of routers, and specify an NET.

2. Set RouterA and RouterC to authenticate Hello packets in specified mode and with the specified password.
3. Check the IS-IS database and the routing table of each router.

## Data Preparation

To complete the configuration, you need the following data:

- Area addresses of Router A, Router B, Router C and Router D
- Levels of Router A, Router B, Router C, and Router D

## Procedure

**Step 1** Configure an IP address for each interface.

This example assumes that you know the configuration method and no details are provided here.

**Step 2** Configure basic IS-IS functions.

# Configure Router A.

```
[RouterA] isis 1
[RouterA-isis-1] is-level level-1
[RouterA-isis-1] network-entity 10.0000.0000.0001.00
[RouterA-isis-1] quit
[RouterA] interface ethernet 1/0/0
[RouterA-Ethernet1/0/0] isis enable 1
[RouterA-Ethernet1/0/0] quit
```

# Configure Router B.

```
[RouterB] isis 1
[RouterB-isis-1] is-level level-1
[RouterB-isis-1] network-entity 10.0000.0000.0002.00
[RouterB-isis-1] quit
[RouterB] interface ethernet 1/0/0
[RouterB-Ethernet1/0/0] isis enable 1
[RouterB-Ethernet1/0/0] quit
```

# Configure Router C.

```
[RouterC] isis 1
[RouterC-isis-1] network-entity 10.0000.0000.0003.00
[RouterC-isis-1] quit
[RouterC] interface ethernet 1/0/0
[RouterC-Ethernet1/0/0] isis enable 1
[RouterC-Ethernet1/0/0] quit
[RouterC] interface ethernet 2/0/0
[RouterC-Ethernet2/0/0] isis enable 1
[RouterC-Ethernet2/0/0] quit
[RouterC] interface ethernet 3/0/0
[RouterC-Ethernet3/0/0] isis enable 1
[RouterC-Ethernet3/0/0] quit
```

# Configure Router D.

```
[RouterD] isis 1
[RouterD-isis-1] is-level level-2
[RouterD-isis-1] network-entity 20.0000.0000.0004.00
[RouterD-isis-1] quit
[RouterD] interface ethernet 2/0/0
[RouterD-Ethernet2/0/0] isis enable 1
[RouterD-Ethernet2/0/0] quit
[RouterD] interface ethernet 1/0/0
[RouterD-Ethernet1/0/0] isis enable 1
```

```
[RouterD-Ethernet1/0/0] quit
```

**Step 3** Configure the authentication mode and password for RouterA and RouterC to authenticate Hello packets.

# Configure RouterA.

```
[RouterA] interface ethernet 1/0/0
[RouterA-Ethernet1/0/0] isis authentication-mode md5 huawei
```

# Configure RouterC.

```
[RouterC] interface ethernet 1/0/0
[RouterC-Ethernet1/0/0] isis authentication-mode md5 huawei
```

**Step 4** Verify the configuration.

# Display the IS-IS LSDB of each router.

```
[RouterA] display isis lsdb
Database information for ISIS(1)

Level-1 Link State Database
LSPID Seq Num Checksum Holdtime Length ATT/P/OL

0000.0000.0001.00-00* 0x00000006 0xbf7d 649 68 0/0/0
0000.0000.0001.01-00* 0x00000002 0xcfb 1157 55 0/0/0
0000.0000.0002.00-00 0x00000003 0xef4d 545 68 0/0/0
0000.0000.0003.00-00 0x00000008 0x3340 582 111 1/0/0
Total LSP(s): 4
*(In TLV)-Leaking Route, *(By LSPID)-Self LSP, +-Self LSP(Extended),
ATT-Attached, P-Partition, OL-Overload
```

```
[RouterB] display isis lsdb
Database information for ISIS(1)

Level-1 Link State Database
LSPID Seq Num Checksum Holdtime Length ATT/P/OL

0000.0000.0001.00-00 0x00000006 0xbf7d 642 68 0/0/0
0000.0000.0002.00-00* 0x00000003 0xef4d 538 68 0/0/0
0000.0000.0002.01-00* 0x00000003 0xef4b 538 68 0/0/0
0000.0000.0003.00-00 0x00000008 0x3340 574 111 1/0/0
Total LSP(s): 4
*(In TLV)-Leaking Route, *(By LSPID)-Self LSP, +-Self LSP(Extended),
ATT-Attached, P-Partition, OL-Overload
```

```
[RouterC] display isis lsdb
Database information for ISIS(1)

Level-1 Link State Database
LSPID Seq Num Checksum Holdtime Length ATT/P/OL

0000.0000.0001.00-00 0x00000006 0xbf7d 638 68 0/0/0
0000.0000.0001.01-00 0x00000002 0xcfb 871 55 0/0/0
0000.0000.0002.00-00 0x00000003 0xef4d 533 68 0/0/0
0000.0000.0003.00-00* 0x00000008 0x3340 569 111 1/0/0
Total LSP(s): 4
*(In TLV)-Leaking Route, *(By LSPID)-Self LSP, +-Self LSP(Extended),
ATT-Attached, P-Partition, OL-Overload
```

```
Level-2 Link State Database
LSPID Seq Num Checksum Holdtime Length ATT/P/OL

0000.0000.0003.00-00* 0x00000008 0x55bb 650 100 0/0/0
0000.0000.0004.00-00 0x00000005 0x6510 629 84 0/0/0
0000.0000.0004.01-00 0x00000001 0xee95 803 55 0/0/0
Total LSP(s): 3
*(In TLV)-Leaking Route, *(By LSPID)-Self LSP, +-Self LSP(Extended),
ATT-Attached, P-Partition, OL-Overload
```

```
[RouterD] display isis lsdb
Database information for ISIS(1)
```

```

 Level-2 Link State Database
LSPID Seq Num Checksum Holdtime Length ATT/P/OL

0000.0000.0003.00-00 0x00000008 0x55bb 644 100 0/0/0
0000.0000.0004.00-00* 0x00000005 0x6510 624 84 0/0/0
0000.0000.0004.01-00* 0x00000001 0xee95 700 55 0/0/0
Total LSP(s) : 3
*(In TLV)-Leaking Route, *(By LSPID)-Self LSP, +-Self LSP(Extended),
 ATT-Attached, P-Partition, OL-Overload

```

# Display the IS-IS routing information of each router. A default route must exist in the Level-1 routing table and the next hop is a Level-1-2 router. A Level-2 router must have all Level-1 and Level-2 routes.

```

[RouterA] display isis route
 Route information for ISIS(1)

 ISIS(1) Level-1 Forwarding Table

IPV4 Destination IntCost ExtCost ExitInterface NextHop Flags

10.1.1.0/24 10 NULL Eth1/0/0 Direct D-/L/-
10.1.2.0/24 20 NULL Eth1/0/0 10.1.1.1 A-/L/-
192.168.0.0/24 20 NULL Eth1/0/0 10.1.1.1 A-/L/-
0.0.0.0/0 10 NULL Eth1/0/0 10.1.1.1 A-/L/-
 Flags: D-Direct, A-Added to URT, L-Advertised in LSPs, S-IGP Shortcut,
 U-Up/Down Bit Set

```

```

[RouterC] display isis route
 Route information for ISIS(1)

 ISIS(1) Level-1 Forwarding Table

IPV4 Destination IntCost ExtCost ExitInterface NextHop Flags

10.1.1.0/24 10 NULL Eth1/0/0 Direct D-/L/-
10.1.2.0/24 10 NULL Eth2/0/0 Direct D-/L/-
192.168.0.0/24 10 NULL Eth3/0/0 Direct D-/L/-
 Flags: D-Direct, A-Added to URT, L-Advertised in LSPs, S-IGP Shortcut,
 U-Up/Down Bit Set

```

```

 ISIS(1) Level-2 Forwarding Table

IPV4 Destination IntCost ExtCost ExitInterface NextHop Flags

10.1.1.0/24 10 NULL Eth1/0/0 Direct D-/L/-
10.1.2.0/24 10 NULL Eth2/0/0 Direct D-/L/-
192.168.0.0/24 10 NULL Eth3/0/0 Direct D-/L/-
172.16.0.0/16 20 NULL Eth3/0/0 192.168.0.2 A-/L/-
 Flags: D-Direct, A-Added to URT, L-Advertised in LSPs, S-IGP Shortcut,
 U-Up/Down Bit Set

```

```

[RouterD] display isis route
 Route information for ISIS(1)

 ISIS(1) Level-2 Forwarding Table

IPV4 Destination IntCost ExtCost ExitInterface NextHop Flags

192.168.0.0/24 10 NULL Eth3/0/0 Direct D-/L/-
10.1.1.0/24 20 NULL Eth3/0/0 192.168.0.1 A-/L/-
10.1.2.0/24 20 NULL Eth3/0/0 192.168.0.1 A-/L/-
172.16.0.0/16 10 NULL Eth2/0/0 Direct D-/L/-
 Flags: D-Direct, A-Added to URT, L-Advertised in LSPs, S-IGP Shortcut,
 U-Up/Down Bit Set

```

----End

## Configuration Files

- Configuration file of Router A

```
#
 sysname RouterA
#
 isis 1
 is-level level-1
 network-entity 10.0000.0000.0001.00
#
 interface Ethernet1/0/0
 ip address 10.1.1.2 255.255.255.0
 isis enable 1
 isis authentication-mode md5 N`C55QK<`=/Q=^Q`MAF4<1!!
#
 return
```

- Configuration file of Router B

```
#
 sysname RouterB
#
 isis 1
 is-level level-1
 network-entity 10.0000.0000.0002.00
#
 interface Ethernet1/0/0
 ip address 10.1.2.2 255.255.255.0
 isis enable 1
#
 return
```

- Configuration file of Router C

```
#
 sysname RouterC
#
 isis 1
 network-entity 10.0000.0000.0003.00
#
 interface Ethernet1/0/0
 ip address 10.1.1.1 255.255.255.0
 isis enable 1
 isis authentication-mode md5 N`C55QK<`=/Q=^Q`MAF4<1!!
#
 interface Ethernet2/0/0
 ip address 10.1.2.1 255.255.255.0
 isis enable 1
#
 interface 3/0/0
 ip address 192.168.0.1 255.255.255.0
 isis enable 1
#
 return
```

- Configuration file of Router D

```
#
 sysname RouterD
#
 isis 1
 is-level level-2
 network-entity 20.0000.0000.0004.00
#
 interface Ethernet1/0/0
 ip address 192.168.0.2 255.255.255.0
 isis enable 1
#
 interface Ethernet2/0/0
 ip address 172.16.1.1 255.255.0.0
 isis enable 1
#
```

return

## 7.18.2 Example for Configuring the DIS Election of IS-IS

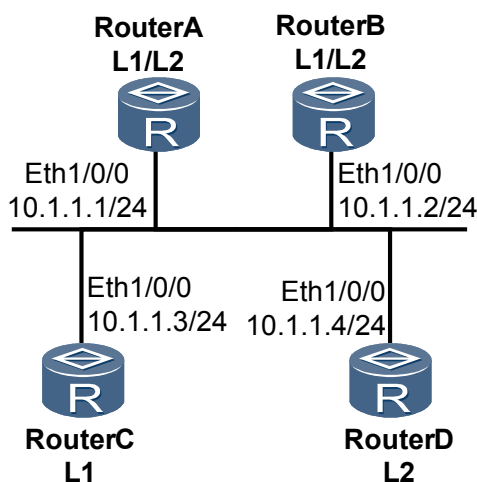
This part provides an example for specifying the DIS on a broadcast network.

### Networking Requirements

As shown in [Figure 7-4](#):

- Router A, Router B, Router C, and Router D run IS-IS to implement interconnection in the network.
- The four routers belong to area 10, and the network type is broadcast (Ethernet).
- Router A and Router B are Level-1-2 routers, Router C is a Level-1 router, and Router D is a Level-2 router.
- The DIS priority of RouterA is 100.
- You can change the DIS priority of the interface to configure Router A as a Level-1-2 DIS.

**Figure 7-4** Configuring the DIS election of IS-IS



### Configuration Roadmap

The configuration roadmap is as follows:

1. Enable IS-IS on each router and specify the network entity to implement interconnection.
2. Check information about IS-IS interfaces on each router in the case of the default preference.
3. Configure the DIS priority of each router.

### Data Preparation

To complete the configuration, you need the following data:

- Area addresses of routerA, routerB, routerC and routerD

- Levels of routerA, routerB, routerC and routerD
- DIS priority of RouterA

## Procedure

### Step 1 Configure an IPv4 address for each interface.

This example assumes that you know the configuration method and no details are provided here.

### Step 2 Check the MAC address of the Eth interface on each router.

# Check the MAC address of Ethernet 1/0/0 on Router A.

```
[RouterA] display arp interface ethernet 1/0/0
IP ADDRESS MAC ADDRESS EXPIRE (M) TYPE INTERFACE VPN-INSTANCE
 VLAN/CEVLAN PVC

10.1.1.1 00e0-fc10-afec I - Eth1/0/0

Total:1 Dynamic:0 Static:0 Interface:1
```

# Check the MAC address of Ethernet1/0/0 on Router B.

```
[RouterB] display arp interface ethernet 1/0/0
IP ADDRESS MAC ADDRESS EXPIRE (M) TYPE INTERFACE VPN-INSTANCE
 VLAN/CEVLAN PVC

10.1.1.2 00e0-fccd-acdf I - Eth1/0/0

Total:1 Dynamic:0 Static:0 Interface:1
```

# Check the MAC address of Ethernet1/0/0 on Router C.

```
[RouterC] display arp interface ethernet 1/0/0
IP ADDRESS MAC ADDRESS EXPIRE (M) TYPE INTERFACE VPN-INSTANCE
 VLAN/CEVLAN PVC

10.1.1.3 00e0-f100-25fe I - Eth1/0/0

Total:1 Dynamic:0 Static:0 Interface:1
```

# Check the MAC address of Ethernet1/0/0 on Router D.

```
[RouterD] display arp interface ethernet 1/0/0
IP ADDRESS MAC ADDRESS EXPIRE (M) TYPE INTERFACE VPN-INSTANCE
 VLAN/CEVLAN PVC

10.1.1.4 00e0-ff1d-305c I - Eth1/0/0

Total:1 Dynamic:0 Static:0 Interface:1
```

### Step 3 Enable IS-IS.

# Configure Router A.

```
[RouterA] isis 1
[RouterA-isis-1] network-entity 10.0000.0000.0001.00
[RouterA-isis-1] quit
[RouterA] interface ethernet 1/0/0
[RouterA-Ethernet1/0/0] isis enable 1
[RouterA-Ethernet1/0/0] quit
```

# Configure Router B.

```
[RouterB] isis 1
[RouterB-isis-1] network-entity 10.0000.0000.0002.00
[RouterB-isis-1] quit
```

```
[RouterB] interface ethernet 1/0/0
[RouterB-Ethernet1/0/0] isis enable 1
[RouterB-Ethernet1/0/0] quit
```

# Configure Router C.

```
[RouterC] isis 1
[RouterC-isis-1] network-entity 10.0000.0000.0003.00
[RouterC-isis-1] is-level level-1
[RouterC-isis-1] quit
[RouterC] interface ethernet 1/0/0
[RouterC-Ethernet1/0/0] isis enable 1
[RouterC-Ethernet1/0/0] quit
```

# Configure Router D.

```
[RouterD] isis 1
[RouterD-isis-1] network-entity 10.0000.0000.0004.00
[RouterD-isis-1] is-level level-2
[RouterD-isis-1] quit
[RouterD] interface ethernet 1/0/0
[RouterD-Ethernet1/0/0] isis enable 1
[RouterD-Ethernet1/0/0] quit
```

# Display the IS-IS neighbors of Router A.

```
[RouterA] display isis peer
Peer information for ISIS(1)

System Id Interface Circuit Id State HoldTime Type PRI

0000.0000.0002 Eth1/0/0 0000.0000.0002.01 Up 9s L1 (L1L2) 64
0000.0000.0003 Eth1/0/0 0000.0000.0002.01 Up 27s L1 64
0000.0000.0002 Eth1/0/0 0000.0000.0004.01 Up 28s L2 (L1L2) 64
0000.0000.0004 Eth1/0/0 0000.0000.0004.01 Up 7s L2 64

Total Peer(s): 4
```

# Display the IS-IS interface of Router A.

```
[RouterA] display isis interface
Interface information for ISIS(1)

Interface Id IPV4.State IPV6.State MTU Type DIS
Eth1/0/0 001 Up Down 1497 L1/L2 No/No
```

# Display the IS-IS interface on Router B.

```
[RouterB] display isis interface
Interface information for ISIS(1)

Interface Id IPV4.State IPV6.State MTU Type DIS
Eth1/0/0 001 Up Down 1497 L1/L2 Yes/No
```

# Display the IS-IS interface of Router D.

```
[RouterD] display isis interface
Interface information for ISIS(1)

Interface Id IPV4.State IPV6.State MTU Type DIS
Eth1/0/0 001 Up Down 1497 L1/L2 No/Yes
```

 **NOTE**

When the default DIS priority is used, the MAC address of the interface on Router B is the largest one among those of Level-1 routers. Router B is thus the DIS of the Level-1 area. The MAC address of interface on Router D is the largest one among those of Level-2 routers. Router D is the DIS of the Level-2 area. The Level-1 and Level-2 pseudo nodes are 0000.0000.0002.01 and 0000.0000.0004.01 respectively.

**Step 4** Configure the DIS priority of Router A.



```
[RouterA] interface ethernet 1/0/0
[RouterA-Ethernet1/0/0] isis dis-priority 100
```

# Display the IS-IS neighbors of Router A.

```
[RouterA] display isis peer
Peer information for ISIS(1)

System Id Interface Circuit Id State HoldTime Type PRI

0000.0000.0002 Eth1/0/0 0000.0000.0001.01 Up 21s L1 (L1L2) 64
0000.0000.0003 Eth1/0/0 0000.0000.0001.01 Up 27s L1 64
0000.0000.0002 Eth1/0/0 0000.0000.0001.01 Up 28s L2 (L1L2) 64
0000.0000.0004 Eth1/0/0 0000.0000.0001.01 Up 30s L2 64

Total Peer(s): 4
```

### Step 5 Verify the configuration.

# Display the IS-IS interface of Router A.

```
[RouterA] display isis interface
Interface information for ISIS(1)

Interface Id IPV4.State IPV6.State MTU Type DIS

Eth1/0/0 001 Up Down 1497 L1/L2 Yes/Yes
```

#### NOTE

After the DIS priority of the IS-IS interface changes, Router A becomes the DIS of the Level-1-2 area instantly and its pseudo node is 0000.0000.0001.01.

# Display the IS-IS neighbors and IS-IS interfaces of Router B.

```
[RouterB] display isis peer
Peer information for ISIS(1)

System Id Interface Circuit Id State HoldTime Type PRI

0000.0000.0001 Eth1/0/0 0000.0000.0001.01 Up 7s L1 (L1L2) 100
0000.0000.0003 Eth1/0/0 0000.0000.0001.01 Up 25s L1 64
0000.0000.0001 Eth1/0/0 0000.0000.0001.01 Up 7s L2 (L1L2) 100
0000.0000.0004 Eth1/0/0 0000.0000.0001.01 Up 25s L2 64

Total Peer(s): 4
[RouterB] display isis interface
Interface information for ISIS(1)

Interface Id IPV4.State IPV6.State MTU Type DIS

Eth1/0/0 001 Up Down 1497 L1/L2 No/No
```

# Display the IS-IS neighbors and interfaces of Router D.

```
[RouterD] display isis peer
Peer information for ISIS(1)

System Id Interface Circuit Id State HoldTime Type PRI

0000.0000.0001 Eth1/0/0 0000.0000.0001.01 Up 9s L2 100
0000.0000.0002 Eth1/0/0 0000.0000.0001.01 Up 28s L2 64

Total Peer(s): 2
[RouterD] display isis interface
Interface information for ISIS(1)

Interface Id IPV4.State IPV6.State MTU Type DIS

Eth1/0/0 001 Up Down 1497 L1/L2 No/No
```

---End

## Configuration Files

- Configuration file of Router A

```
#
 sysname RouterA
#
 isis 1
 network-entity 10.0000.0000.0001.00
#
 interface Ethernet1/0/0
 ip address 10.1.1.1 255.255.255.0
 isis enable 1
 isis dis-priority 100
#
 return
```

- Configuration file of Router B

```
#
 sysname RouterB
#
 isis 1
 network-entity 10.0000.0000.0002.00
#
 interface Ethernet1/0/0
 ip address 10.1.1.2 255.255.255.0
 isis enable 1
#
 return
```

- Configuration file of Router C

```
#
 sysname RouterC
#
 isis 1
 is-level level-1
 network-entity 10.0000.0000.0003.00
#
 interface Ethernet1/0/0
 ip address 10.1.1.3 255.255.255.0
 isis enable 1
#
 return
```

- Configuration file of Router D

```
#
 sysname RouterD
#
 isis 1
 is-level level-2
 network-entity 10.0000.0000.0004.00
#
 interface Ethernet1/0/0
 ip address 10.1.1.4 255.255.255.0
 isis enable 1
#
 return
```

### 7.18.3 Example for Configuring Basic IS-IS IPv6 Functions

This part provides an example for interconnecting IPv6 networks through IS-IS.

#### Networking Requirements

As shown in [Figure 7-5](#):

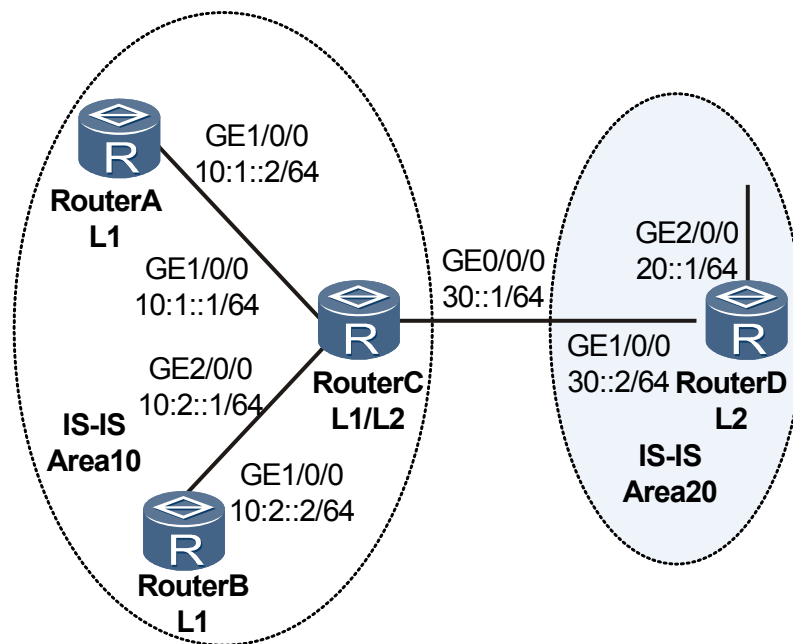
- Router A, Router B, Router C, and Router D belong to the same AS. They are interconnected through IS-IS in the IPv6 network.
- Router A, Router B, and Router C belong to area 10. Router D belongs to area 20.
- Router A and Router B are Level-1 routers. Router C is a Level-1-2 router. Router D is a Level-2 router.



**NOTE**

AR150/200 is RouterA, or RouterB.

**Figure 7-5** Networking diagram of basic IS-IS IPv6 feature



## Configuration Roadmap

The configuration roadmap is as follows:

1. Enable the capability of IPv6 forwarding on each router.
2. Configure an IPv6 address for each interface.
3. Enable IS-IS on each router.
4. Configure the level.
5. Specify the network entity.

## Data Preparation

To complete the configuration, you need the following data:

- IPv6 address of each interface on Router A, Router B, Router C, and Router D
- Area numbers of Router A, Router B, Router C, and Router D
- Levels of Router A, Router B, Router C, and Router D

## Procedure

- Step 1** Enable the capability of IPv6 forwarding, and configure IPv6 address for each interface. Take the display on Router A as an example. The configurations of Router B, Router C and Router D are similar to that of Router A. The detailed configurations are not mentioned here.

```
<Huawei> system-view
[Huawei] sysname RouterA
[RouterA] ipv6
[RouterA] interface ethernet 1/0/0
[RouterA-Ethernet1/0/0] ipv6 enable
[RouterA-Ethernet1/0/0] ipv6 address 10:1::2/64
```

- Step 2** Configure IS-IS.

# Configure Router A.

```
[RouterA] isis 1
[RouterA-isis-1] is-level level-1
[RouterA-isis-1] network-entity 10.0000.0000.0001.00
[RouterA-isis-1] ipv6 enable
[RouterA-isis-1] quit
[RouterA] interface ethernet 1/0/0
[RouterA-Ethernet1/0/0] isis ipv6 enable 1
[RouterA-Ethernet1/0/0] quit
```

# Configure Router B.

```
[RouterB] isis 1
[RouterB-isis-1] is-level level-1
[RouterB-isis-1] network-entity 10.0000.0000.0002.00
[RouterB-isis-1] ipv6 enable
[RouterB-isis-1] quit
[RouterB] interface ethernet 1/0/0
[RouterB-Ethernet1/0/0] isis ipv6 enable 1
[RouterB-Ethernet1/0/0] quit
```

# Configure Router C.

```
[RouterC] isis 1
[RouterC-isis-1] network-entity 10.0000.0000.0003.00
[RouterC-isis-1] ipv6 enable
[RouterC-isis-1] quit
[RouterC] interface ethernet 1/0/0
[RouterC-Ethernet1/0/0] isis ipv6 enable 1
[RouterC-Ethernet1/0/0] quit
[RouterC] interface ethernet 2/0/0
[RouterC-Ethernet2/0/0] isis ipv6 enable 1
[RouterC-Ethernet2/0/0] quit
[RouterC] interface ethernet 0/0/0
[RouterC-Ethernet0/0/0] isis ipv6 enable 1
[RouterC-Ethernet0/0/0] isis circuit-level level-2
[RouterC-Ethernet0/0/0] quit
```

# Configure Router D.

```
[RouterD] isis 1
[RouterD-isis-1] is-level level-2
[RouterD-isis-1] network-entity 20.0000.0000.0004.00
[RouterD-isis-1] ipv6 enable
[RouterD-isis-1] quit
[RouterD] interface ethernet 1/0/0
[RouterD-Ethernet1/0/0] isis ipv6 enable 1
[RouterD-Ethernet1/0/0] quit
[RouterD] interface ethernet 2/0/0
[RouterD-Ethernet2/0/0] isis ipv6 enable 1
[RouterD-Ethernet2/0/0] quit
```

**Step 3** Verify the configuration.

# Display the IS-IS routing table of Router A.

```
[RouterA] display isis route
 Route information for ISIS(1)

 ISIS(1) Level-1 Forwarding Table

IPV4 Destination IntCost ExtCost ExitInterface NextHop Flags

0.0.0.0/0 10 NULL
IPV6 Dest. ExitInterface NextHop Cost Flags

::/0 Eth1/0/0 FE80::A83E:0:3ED2:1 10 A/-/-
10:1::/64 Eth1/0/0 Direct 10 D/L/-
10:2::/64 Eth1/0/0 FE80::A83E:0:3ED2:1 20 A/-/-
 Flags: D-Direct, A-Added to URT, L-Advertised in LSPs, S-IGP Shortcut,
 U-Up/Down Bit Set
```

# Display the IS-IS neighbors of Router C.

```
[RouterC] display isis peer verbose
 Peer information for ISIS(1)

System Id Interface Circuit Id State HoldTime Type PRI

0000.0000.0001 Eth1/0/0 0000000001 Up 24s L1 --
MT IDs supported : 0 (UP)
Local MT IDs : 0
Area Address(es) : 10
Peer IPv6 Address(es): FE80::996B:0:9419:1
Uptime : 00:44:43
Adj Protocol : IPV6
Restart Capable : YES
Suppressed Adj : NO
0000.0000.0002 Eth2/0/0 0000000001 Up 28s L1 --
MT IDs supported : 0 (UP)
Local MT IDs : 0
Area Address(es) : 10
Peer IPv6 Address(es): FE80::DC40:0:47A9:1
Uptime : 00:46:13
Adj Protocol : IPV6
Restart Capable : YES
Suppressed Adj : NO
0000.0000.0004 Eth0/0/0 0000000001 Up 24s L2 --
MT IDs supported : 0 (UP)
Local MT IDs : 0
Area Address(es) : 20
Peer IPv6 Address(es): FE80::F81D:0:1E24:2
Uptime : 00:53:18
Adj Protocol : IPV6
Restart Capable : YES
Suppressed Adj : NO
Total Peer(s) : 3
```

# Display the IS-IS LSDB of Router C.

```
[RouterC] display isis lsdb verbose
 Database information for ISIS(1)

 Level-1 Link State Database
LSPID Seq Num Checksum Holdtime Length ATT/P/OL

0000.0000.0001.00-00 0x0000000c 0x4e06 1117 113 0/0/0
SOURCE 0000.0000.0001.00
NLPID IPV6
AREA ADDR 10
INTF ADDR V6 10:1::2
Topology Standard
NBR ID 0000.0000.0003.00 COST: 10
```

```

 IPV6 10:1::/64 COST: 10
0000.0000.0002.00-00 0x00000009 0x738c 1022 83 0/0/0
SOURCE 0000.0000.0002.00
NLPID IPV6
AREA ADDR 10
INTF ADDR V6 10:2::2
Topology Standard
NBR ID 0000.0000.0003.00 COST: 10
IPV6 10:2::/64 COST: 10
0000.0000.0003.00-00* 0x00000020 0x6b10 771 140 1/0/0
SOURCE 0000.0000.0003.00
NLPID IPV6
AREA ADDR 10
INTF ADDR V6 30::1
INTF ADDR V6 10:2::1
INTF ADDR V6 10:1::1
Topology Standard
NBR ID 0000.0000.0002.00 COST: 10
NBR ID 0000.0000.0001.00 COST: 10
IPV6 10:2::/64 COST: 10
IPV6 10:1::/64 COST: 10
Total LSP(s): 5
 *(In TLV)-Leaking Route, *(By LSPID)-Self LSP, +-Self LSP(Extended),
 ATT-Attached, P-Partition, OL-Overload
Level-2 Link State Database
LSPID Seq Num Checksum Holdtime Length ATT/P/OL

0000.0000.0003.00-00* 0x00000017 0x61b4 771 157 0/0/0
SOURCE 0000.0000.0003.00
NLPID IPV6
AREA ADDR 10
INTF ADDR V6 30::1
INTF ADDR V6 10:2::1
INTF ADDR V6 10:1::1
Topology Standard
NBR ID 0000.0000.0004.00 COST: 10
IPV6 30::/64 COST: 10
IPV6 10:2::/64 COST: 10
IPV6 10:1::/64 COST: 10
0000.0000.0004.00-00 0x0000000b 0x6dfa 1024 124 0/0/0
SOURCE 0000.0000.0004.00
NLPID IPV6
AREA ADDR 20
INTF ADDR V6 30::2
INTF ADDR V6 20::1
Topology Standard
NBR ID 0000.0000.0003.00 COST: 10
NBR ID 0000.0000.0005.00 COST: 10
IPV6 30::/64 COST: 10
IPV6 20::/64 COST: 10
Total LSP(s): 3
 *(In TLV)-Leaking Route, *(By LSPID)-Self LSP, +-Self LSP(Extended),
 ATT-Attached, P-Partition, OL-Overload

```

----End

## Configuration Files

- Configuration file of Router A

```

#
sysname RouterA
#
ipv6
#
isis 1
 is-level level-1
 network-entity 10.0000.0000.0001.00
#

```

```
 ipv6 enable topology standard
#
interface Ethernet1/0/0
 ipv6 enable
 ipv6 address 10:1::2/64
 isis ipv6 enable 1
#
return
```

- Configuration file of Router B

```
#
sysname RouterB
#
ipv6
#
isis 1
 is-level level-1
 network-entity 10.0000.0000.0002.00
#
 ipv6 enable topology standard
#
interface Ethernet1/0/0
 ipv6 enable
 ipv6 address 10:2::2/64
 isis ipv6 enable 1
#
return
```

- Configuration file of Router C

```
#
sysname RouterC
#
ipv6
#
isis 1
 network-entity 10.0000.0000.0003.00
#
 ipv6 enable topology standard
#
interface Ethernet0/0/0
 ipv6 enable
 ipv6 address 30::1/64
 isis ipv6 enable 1
 isis circuit-level level-2
#
interface Ethernet1/0/0
 ipv6 enable
 ipv6 address 10:1::1/64
 isis ipv6 enable 1
#
interface Ethernet2/0/0
 ipv6 enable
 ipv6 address 10:2::1/64
 isis ipv6 enable 1
#
return
```

- Configuration file of Router D

```
#
sysname RouterD
#
ipv6
#
isis 1
 is-level level-2
 network-entity 20.0000.0000.0004.00
#
 ipv6 enable topology standard
#
interface Ethernet2/0/0
```

```
ipv6 enable
ipv6 address 20::1/64
isis ipv6 enable 1
#
interface Ethernet1/0/0
ipv6 enable
ipv6 address 30::2/64
isis ipv6 enable 1
#
return
```

## 7.18.4 Example for Configuring IS-IS Fast Convergence

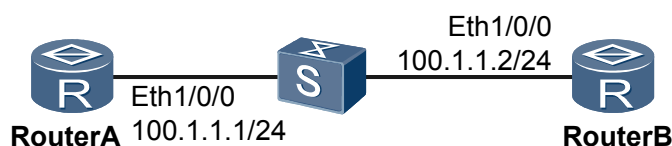
This part provides an example for implementing fast route convergence by adjusting IS-IS timers.

### Networking Requirements

As shown in [Figure 7-6](#):

- Router A and Router B run IS-IS.
- Router A and Router B belong to area 10. They are Level-2 routers.
- A Layer 2 switch, which need not be configured, connects Router A and Router B.

**Figure 7-6** Networking diagram of IS-IS fast convergence



### Configuration Roadmap

The configuration roadmap is as follows:

1. Enable basic IS-IS functions on each router.
2. Enable BFD on Router A and Router B.
3. Set the time parameters of fast convergence on Router A and Router B.

### Data Preparation

To configure IS-IS fast convergence, you need the following data:

- Levels and area addresses of the two routers
- Time parameters of fast convergence

### Procedure

**Step 1** Configure an IP address for each interface.

This example assumes that you know the configuration method and no details are provided here.



**Step 2** Configure basic IS-IS functions.

# Configure Router A.

```
[RouterA] isis 1
[RouterA-isis-1] is-level level-2
[RouterA-isis-1] network-entity 10.0000.0000.0001.00
[RouterA-isis-1] quit
[RouterA] interface ethernet 1/0/0
[RouterA-Ethernet1/0/0] isis enable 1
[RouterA-Ethernet1/0/0] quit
```

# Configure Router B.

```
[RouterB] isis 1
[RouterB-isis-1] is-level level-2
[RouterB-isis-1] network-entity 10.0000.0000.0002.00
[RouterB-isis-1] quit
[RouterB] interface ethernet 1/0/0
[RouterB-Ethernet1/0/0] isis enable 1
[RouterB-Ethernet1/0/0] quit
```

**Step 3** Configure BFD.

# Configure Router A.

```
[RouterA] bfd
[RouterA-bfd] quit
[RouterA] bfd atob bind peer-ip 100.1.1.2 interface ethernet 1/0/0
[RouterA-bfd-session-atob] discriminator local 1
[RouterA-bfd-session-atob] discriminator remote 2
[RouterA-bfd-session-atob] commit
[RouterA-bfd-session-atob] quit
[RouterA] interface ethernet 1/0/0
[RouterA-Ethernet1/0/0] isis bfd static
[RouterA-Ethernet1/0/0] quit
```

# Configure Router B.

```
[RouterB] bfd
[RouterB-bfd] quit
[RouterB] bfd btoa bind peer-ip 100.1.1.1 interface ethernet 1/0/0
[RouterB-bfd-session-btoa] discriminator local 2
[RouterB-bfd-session-btoa] discriminator remote 1
[RouterB-bfd-session-btoa] commit
[RouterB-bfd-session-btoa] quit
[RouterB] interface ethernet 1/0/0
[RouterB-Ethernet1/0/0] isis bfd static
[RouterB-Ethernet1/0/0] quit
```

**Step 4** Set the time parameters of fast convergence.

# Configure Router A.

```
[RouterA] isis 1
[RouterA-isis-1] flash-flood
[RouterA-isis-1] timer spf 1 20 100
[RouterA-isis-1] timer lsp-generation 1 1 120
[RouterA-isis-1] quit
```

# Configure Router B.

```
[RouterB] isis 1
[RouterB-isis-1] flash-flood
[RouterB-isis-1] timer spf 1 20 100
[RouterB-isis-1] timer lsp-generation 1 1 120
[RouterB-isis-1] quit
```

 **NOTE**

- In IS-IS, if LSDB changes, routes are calculated and then a new LSP is generated to report this change. Frequent route calculations consume lots of system resources and degrades the system performance. Delaying SPF calculation, generating a new LSP time, and LSP fast flooding improves the efficiency in route calculation and reduces the consumption of system resources.
- Using the **flash-flood** command, you can enable LSP fast flooding to speed up the convergence of an IS-IS network.
- Run the **timer spf** command to set the interval of the SPF calculation. By default, the interval is 5 seconds.
- Run the **timer lsp-generation** command to set the delay for generating an LSP. By default, the delay is 2 seconds.

**Step 5** Verify the configuration.

# Run the **shutdown** command on Eth 1/0/0 of Router B to simulate the link in the Down state.

```
[RouterB] interface ethernet 1/0/0
[RouterB-Ethernet1/0/0] shutdown
```

# View the information about neighbors of Router A.

```
<RouterA> display isis peer
```

Information about neighbors of Router A does not exist.

When BFD detects that the link goes Down, it notifies the route management (RM) module immediately. IS-IS then deletes neighbors immediately and triggers the route calculation. This results in the fast convergence of the network.

----End

## Configuration Files

- Configuration file of Router A

```
#
sysname RouterA
#
bfd
#
isis 1
is-level level-2
timer lsp-generation 1 1 120 level-1
timer lsp-generation 1 1 120 level-2
flash-flood level-1
flash-flood level-2
network-entity 10.0000.0000.0001.00
timer spf 1 20 100
#
interface Ethernet1/0/0
ip address 100.1.1.1 255.255.255.0
isis enable 1
isis bfd static
#
bfd btoa bind peer-ip 100.1.1.2 interface Ethernet1/0/0
discriminator local 1
discriminator remote 2
commit
#
return
```

- Configuration file of Router B

```
#
sysname RouterB
#
```

```
bfd
#
isis 1
 is-level level-2
 timer lsp-generation 1 1 120 level-1
 timer lsp-generation 1 1 120 level-2
 flash-flood level-1
 flash-flood level-2
 network-entity 10.0000.0000.0002.00
 timer spf 1 20 100
#
interface Ethernet1/0/0
 ip address 100.1.1.2 255.255.255.0
 isis enable 1
 isis bfd static
#
bfd toa bind peer-ip 100.1.1.1 interface Ethernet1/0/0
 discriminator local 2
 discriminator remote 1
 commit
#
return
```

# 8 BGP Configuration

---

## About This Chapter

BGP is used between ASs to transmit routing information on large-scale and complex networks.

### [8.1 BGP Overview](#)

BGP is mainly used to control route transmission and select the optimal route.

### [8.2 BGP Features Supported by the AR150/200](#)

The system supports various BGP features, including route summarization, peer group, route reflector, confederation, community, MP-BGP, BGP ORF, BGP Tracking, route dampening, load balancing, path MTU auto discovery, BGP next hop delayed response, BFD for BGP BGP GR, and BGP security.

### [8.3 Configuring Basic BGP Functions](#)

Configuring basic BGP functions is the prerequisite to building a BGP network.

### [8.4 Configuring BGP Route Attributes](#)

BGP has many route attributes. Configuring route attributes can change route selection results.

### [8.5 Configuring BGP to Advertise Routes](#)

BGP is used to transmit routing information. BGP advertises only the wanted routes after filtering routes to be advertised, and modifies route attributes to direct network traffic.

### [8.6 Configuring BGP to Receive Routes](#)

BGP is used to transmit routing information. BGP can filter received routes to accept only the expected routes, and can modify route attributes to direct network traffic.

### [8.7 Configuring BGP Route Aggregation](#)

Configuring BGP Route Aggregation on a device can reduce the sizes of routing tables on the peers of the device.

### [8.8 Configuring BGP Peer Groups](#)

Configuring BGP peer groups simplifies the BGP network configuration and improves the route advertisement efficiency.

### [8.9 Configuring BGP Route Reflectors](#)

Deploying BGP RRs allows IBGP peers to communicate without establishing full-mesh connections between them. Using BGP RRs simplifies network configurations and improves route advertisement efficiency.

### 8.10 Configuring a BGP Confederation

BGP confederations can be configured on a large BGP network to reduce the number of IBGP connections and simplify routing policy management, increasing route advertisement efficiency.

### 8.11 Configuring BGP Community Attributes

Community attributes are used to simplify routing policy management.

### 8.12 Configuring Prefix-based BGP ORF

Prefix-based BGP ORF is used to enable a BGP device to send to its BGP peer a set of routing policies that can be used by its peer to filter out unwanted routes during route advertisement.

### 8.13 Configuring to Adjust the BGP Network Convergence Speed

You can adjust the BGP network convergence speed by adjusting BGP peer connection parameters to adapt to changes on large-scale networks.

### 8.14 Configuring BGP Route Dampening

BGP route dampening can be configured to suppress unstable routes.

### 8.15 Configuring a BGP Device to Send a Default Route to Its Peer

After a BGP device is configured to send a default route to its peer, the BGP device sends a default route with the local address as the next-hop address to a specified peer, regardless of whether there are default routes in the local routing table. This greatly reduces the number of routes on the network.

### 8.16 Configuring BGP Load Balancing

Configuring BGP load balancing better utilizes network resources and reduces network congestion.

### 8.17 Configuring Path MTU Auto Discovery

Path MTU auto discovery allows BGP to discover the smallest MTU value on a path to ensure that BGP messages satisfy the path MTU requirement. This function improves transmission efficiency and BGP performance.

### 8.18 Configuring the BGP Next Hop Delayed Response

Configuring the BGP next hop delayed response can minimize traffic loss during route changes.

### 8.19 Configuring BFD for BGP

BFD for BGP speeds up fault detection and therefore increases the route convergence speed.

### 8.20 Configuring BGP GR

BGP GR can be configured to avoid traffic interruption due to protocol restart.

### 8.21 Configuring BGP Security

Authentication can be implemented during the establishment of a TCP connection to enhance BGP security.

### 8.22 Maintaining BGP

Maintaining BGP involves resetting a BGP connection and clearing BGP statistics.

### 8.23 Configuration Examples

BGP configuration examples explain networking requirements, networking diagram, configuration notes, configuration roadmap, and configuration procedure.

## 8.1 BGP Overview

BGP is mainly used to control route transmission and select the optimal route.

The Border Gateway Protocol (BGP) is a dynamic routing protocol used between Autonomous Systems (ASs). BGP-1 (defined in RFC 1105), BGP-2 (defined in RFC 1163), and BGP-3 (defined in RFC 1267) are three earlier-released versions of BGP. The current BGP version is BGP-4 defined in RFC 4271.

As an exterior routing protocol on the Internet, BGP is widely used among Internet Service Providers (ISPs).

### NOTE

Unless otherwise stated, BGP stated in this document refers to BGP-4.

Characteristics of BGP are as follows:

- Different from the Internal Gateway Protocol (IGP) such as the Open Shortest Path First (OSPF) and Routing Information Protocol (RIP), BGP is an Exterior Gateway Protocol (EGP), which controls route advertisement and selects the optimal route between ASs rather than discover or calculate routes.
- BGP uses the Transport Control Protocol (TCP) with the port number being 179 as the transport layer protocol. The reliability of BGP is thus enhanced.
- BGP supports Classless Inter-Domain Routing (CIDR).
- BGP transmits only the updated routes when routes are being updated. This reduces the bandwidth occupied by BGP for route distribution. Therefore, BGP is applicable to the Internet where a large number of routes are transmitted.
- BGP eliminates routing loops by adding AS path information to BGP routes.
- BGP provides rich routing policies to flexibly select and filter routes.
- BGP can be easily extended and adapt to the development of networks.

BGP runs on the router in either of the following modes:

- Internal BGP (IBGP)
- External BGP (EBGP)

When BGP runs within an AS, it is called IBGP. When BGP runs between ASs, it is called EBGP.

## 8.2 BGP Features Supported by the AR150/200

The system supports various BGP features, including route summarization, peer group, route reflector, confederation, community, MP-BGP, BGP ORF, BGP Tracking, route dampening, load balancing, path MTU auto discovery, BGP next hop delayed response, BFD for BGP BGP GR, and BGP security.

### Main Route Attributes



- Origin attribute
- AS\_Path attribute

- Next\_Hop attribute
- Multi-Exit-Discriminator (MED) attribute
- Local\_Pref attribute
- Community attribute

## Principles of Route Selection

On the AR150/200, when there are multiple active routes to the same destination, BGP selects routes according to the following principles:

1. Prefers the route with the highest PreVal.  
PrefVal is a Huawei-specific parameter. It is valid only on the device where it is configured.
2. Prefers the route with the highest Local\_Pref.  
A route without Local\_Pref is considered to have had the value set by using the **default local-preference** command or to have a value of 100 by default.
3. Prefers a locally originated route. A locally originated route takes precedence over a route learned from a peer.  
Locally originated routes include routes imported by using the **network** command or the **import-route** command, manually aggregated routes, and automatically summarized routes.
  - a. A summarized route is preferred. A summarized route takes precedence over a non-summarized route.
  - b. A route obtained by using the **aggregate** command is preferred over a route obtained by using the **summary automatic** command.
  - c. A route imported by using the **network** command is preferred over a route imported by using the **import-route** command.
4. Prefers the route with the shortest AS\_Path.
  - The AS\_CONFED\_SEQUENCE and AS\_CONFED\_SET are not included in the AS\_Path length.
  - An AS\_SET counts as 1, no matter how many ASs are in the set.
  - After the **bestroute as-path-ignore** command is run, the AS\_Path attributes of routes are not compared in the route selection process.
5. Prefers the route with the highest Origin type. IGP is higher than EGP, and EGP is higher than Incomplete.
6. Prefers the route with the lowest Multi Exit Discriminator (MED).
  - The MEDs of only routes from the same AS but not a confederation sub-AS are compared. MEDs of two routes are compared only when the first AS number in the AS\_SEQUENCE (excluding AS\_CONFED\_SEQUENCE) is the same for the two routes.
  - A route without any MED is assigned a MED of 0, unless the **bestroute med-none-as-maximum** command is run. If the **bestroute med-none-as-maximum** command is run, the route is assigned the highest MED of 4294967295.
  - After **compare-different-as-med** command is run, the MEDs in routes sent from peers in different ASs are compared. Do not use this command unless it is confirmed that different ASs use the same IGP and route selection mode. Otherwise, a loop may occur.

- If the **bestroute med-confederation** command is run, MEDs are compared for routes that consist only of AS\_CONFED\_SEQUENCE. The first AS number in the AS\_CONFED\_SEQUENCE must be the same for the routes.
  - After the **deterministic-med** command is run, routes are not selected in the sequence in which routes are received.
7. Prefers EBGP routes over IBGP routes.
- EBGP is higher than IBGP, IBGP is higher than LocalCross, and LocalCross is higher than RemoteCross.
- If the ERT of a VPNv4 route in the routing table of a VPN instance on a PE matches the IRT of another VPN instance on the PE, the VPNv4 route will be added to the routing table of the second VPN instance. This is called LocalCross. If the ERT of a VPNv4 route from a remote PE is learned by the local PE and matches the IRT of a VPN instance on the local PE, the VPNv4 route will be added to the routing table of that VPN instance. This is called RemoteCross.
8. Prefers the route with the lowest IGP metric to the BGP next hop.
-  **NOTE**
- Assume that load balancing is configured. If the preceding rules are the same and there are multiple external routes with the same AS\_Path, load balancing will be performed based on the number of configured routes.
9. Prefers the route with the shortest Cluster\_List.
10. Prefers the route advertised by the router with the smallest router ID.
-  **NOTE**
- If routes carry the Originator\_ID, the originator ID is substituted for the router ID during route selection. The route with the smallest Originator\_ID is preferred.
11. Prefers the route learned from the peer with the smallest address if the IP addresses of peers are compared in the route selection process.

## Policies for BGP Route Advertisement

On the AR150/200, BGP advertises routes based on the following policies:

- When there are multiple active routes, the BGP speaker advertises only the optimal route to its peer.
- The BGP speaker advertises only the preferred routes to its peer.
- The BGP speaker advertises the routes learned from EBGP peers to all BGP peers (including EBGP peers and IBGP peers) except the peers that advertise these routes.
- The BGP speaker does not advertise the routes learned from IBGP peers to its IBGP peers.
- The BGP speaker advertises the routes learned from IBGP peers to its EBGP peers.
- The BGP speaker advertises all preferred BGP routes to the new peers when peer relationships are established.

## Routing Selection Policies for Load Balancing

In BGP, the next-hop address of a generated route may not be the address of the peer that is directly connected to the local router. One common scenario is that the next hop is not changed when a route is advertised between IBGP peers. Therefore, before forwarding a packet, the router must find a directly reachable address, through which the packet can reach the next hop specified in the routing table. In this process, the route to the directly reachable address is called



a dependent route. BGP routes depend on these dependent routes for packet forwarding. The process of finding a dependent route based on the next-hop address is called route iteration.

The AR150/200 supports iteration-based BGP load balancing. If load balancing is configured for a dependent route (assume that there are three next-hop addresses), BGP generates the same number of next-hop addresses to forward packets. BGP load balancing based on iteration does not need to be configured by using commands. This feature is always enabled on the AR150/200.

BGP load balancing is different from IGP load balancing in the following implementation methods:

- In IGPs, if there are different routes to the same destination address, an IGP calculates metrics of these routes based on its own routing algorithm and performs load balancing among the routes with the same metric.
- BGP does not have a routing algorithm. Therefore, BGP cannot determine whether to perform load balancing among routes based on explicit metrics. BGP, however, contains many route attributes, which have different priorities in route selection policies. Therefore, BGP performs load balancing according to route selection policies. That is, load balancing is performed according to the configured maximum number of equal-cost routes only when all the routes have the same high preference.

 **NOTE**

- By default, BGP performs load balancing only among the routes with the same AS\_Path attribute. You can use the **bestroute as-path-ignore** command to configure BGP not to compare the AS\_Path attribute of routes when performs load balancing.
- BGP load balancing is also applicable between ASs in a confederation.

## Route Summarization

On a large-scale network, the BGP routing table is large. You can configure route summarization to reduce the size of the routing table.

Route summarization is the process of consolidating multiple routes into one single advertisement. After route summarization is configured, BGP advertises only the summarized route rather than all specific routes to its peers.

The AR150/200 supports automatic summarization and manual summarization. Manual summarization can be used to control attributes of the summarized route and determine whether to advertise its specific routes.

## Synchronization Between IBGP and IGP

Synchronization between IBGP and IGP is a method of preventing external routes from being imported by error.

If the synchronization function is configured, the IGP routing table is examined before an IBGP route is added to the routing table and advertised to EBGP peers. The IBGP route is added to the routing table and advertised to EBGP peers only when the IGP knows this IBGP route.

The synchronization function can be disabled in the following situations:

- The local AS is not a transit AS.
- Full-mesh IBGP connections are established between all routers in the local AS.

 **NOTE**

In the AR150/200, the synchronization function is disabled by default.

## Peer Group

A peer group is a group of peers with the same policies. After a peer is added to a peer group, it inherits the configurations of this peer group. When the configurations of the peer group are changed, the configurations of peers in the peer group are changed accordingly.

On a large-scale BGP network, there are a large number of peers and most of them have the same policies. To configure these peers, you have to repeatedly use some commands. In such a case, you can simplify configurations by using the peer group.

Adding many peers to a peer group also speeds up route advertisement.

## Route Reflector

To ensure the routing synchronization between IBGP peers, you need to establish full-mesh connections between the IBGP peers. If there are  $n$  routers in an AS,  $n(n-1)/2$  IBGP connections need to be established. When there are a large number of IBGP peers, network resources and CPU resources are greatly consumed.

To solve this problem, route reflection is introduced. In an AS, one router functions as a route reflector (RR) and other routers serve as the clients of the RR. The clients establish IBGP connections with the RR. The RR transmits or reflects routes among clients, and the clients do not need to establish BGP connections.

A BGP router that is neither an RR nor a client is a non-client. Full-mesh connections must be established between non-clients and an RR, and between all non-clients.

## Confederation

Confederation is another method of dealing with increasing IBGP connections in an AS. It divides an AS into several sub-ASs. IBGP connections are established between IBGP peers within each sub-AS, and EBGP connections are established between sub-ASs.

For BGP speakers outside a confederation, sub-ASs in the same confederation are invisible. External devices do not need to know the topology of each sub-AS. The confederation ID is the AS number that is used to identify the entire confederation.

The confederation has disadvantages. That is, if the router needs to be reconfigured in a confederation, the logical topology changes accordingly.

On a large-scale BGP network, the RR and confederation can be used together.

## Community

The community attribute is a route attribute. It is transmitted between BGP peers and is not restricted by the AS. A peer group allows a group of peers to share the same policies, whereas the community allows a group of BGP routers in multiple ASs to share the same policies.

Before a BGP router advertises the route with the community attribute to other peers, it can change the community attribute of this route.

Besides well-known communities, you can use a community filter to filter self-defined extended community attributes to control routing policies in a more flexible manner.

## Introduction to MP-BGP

Traditional BGP-4 manages only IPv4 unicast routing information and has limitations in inter-AS routing when used in the applications of other network layer protocols such as IPv6.

To support multiple network layer protocols, the Internet Engineering Task Force (IETF) extends BGP-4 to Multiprotocol Extensions for BGP-4 (MP-BGP). The current MP-BGP standard is RFC 2858 (Multiprotocol Extensions for BGP-4).

MP-BGP is forward compatible. That is, the routers that support MP-BGP can communicate with the routers that do not support MP-BGP.

## Extended Attributes of MP-BGP

Among BGP-4 packets, an Update packet carries three IPv4-related attributes: Network Layer Reachability Information (NLRI), Next\_Hop, and Aggregator. The Aggregator attribute contains the IP address of the BGP speaker that performs route summarization.

To support multiple types of network layer protocols, BGP-4 needs to carry network layer protocol information in the NLRI attribute and Next\_Hop attribute. MP-BGP introduces two new route attributes:

- Multiprotocol Reachable NLRI (MP\_REACH\_NLRI): It is used to advertise reachable routes and next hops.
- Multiprotocol Unreachable NLRI (MP\_UNREACH\_NLRI): It is used to withdraw unreachable routes.

The two new attributes are optional non-transitive. Therefore, the BGP speakers that do not support the multiprotocol capability will ignore the two attributes, and do not advertise the information to peers.

## Address Family

BGP uses address families to distinguish different network layer protocols. For the values of address families, see RFC 1700 (Assigned Numbers). The AR150/200 supports multiple MP-BGP extensions, such as VPN extension and IPv6 extension, which are configured in their respective address family views.

### NOTE

This chapter does not describe the commands related to a specific application in the MP-BGP address family view.

For the configuration in the BGP IPv6 address family view, see the chapter "BGP4+ Configuration." For the application of MP-BGP in multicast, see the chapter "MBGP Configuration" in the *Huawei AR150&200 Series Enterprise Routers Configuration Guide - IP Multicast*.

For the configuration in the BGP VPNv4 address family view, BGP VPN instance address family view, and BGP L2VPN address family view, see the *Huawei AR150&200 Series Enterprise Routers Configuration Guide - VPN*.

## BGP ORF

BGP Outbound Route Filtering (ORF) is used to implement on-demand BGP route distribution. A device configured with BGP ORF filters BGP routes based on an export policy (only IP prefix list can be used in the export policy currently) before sending them to a remote peer. This export policy is provided by the remote peer. This enables the local device to send only routes required by the remote peer and prevents unnecessary route distribution. The local device does not need

to maintain an export policy for each BGP peer. This greatly reduces the load of the local device and configuration load.

## BGP Tracking

BGP tracking speeds up network convergence by adjusting the interval between peer unreachability discovery and connection interruption. It is easy to deploy and has a good extensibility.

## Route Dampening

Route dampening is a method of solving the problem of route instability. Route instability is reflected by route flapping. That is, a route in the routing table disappears and appears repeatedly.

If route flapping occurs, a routing protocol sends an Update message to its peers. After receiving this Update message, the peers recalculate routes and modify their routing tables. Frequent route flapping consumes a lot of bandwidth and CPU resources, even affecting the normal operation of the network.

In most cases, BGP is applicable to complex networks where routes change frequently. To avoid the impact of frequent route flapping, BGP suppresses unstable routes by using route dampening.

## BGP Path MTU Auto Discovery

Path MTU auto discovery discovers the smallest MTU on a path to ensure that BGP message transmission meets the path MTU requirement. This can improve the efficiency of BGP message transmission.

## BGP Next Hop Delayed Response

BGP next hop delayed response can be used to speed up BGP route convergence and minimize traffic loss when the upstream path of a PE connected to an RR changes.

## BFD for BGP

The AR150/200 supports Bidirectional Forwarding Detection (BFD) in IPv4 to provide fast link failure detection for BGP peer relationship.

BFD can rapidly detect faults on the links between BGP peers and report the faults to BGP, thus implementing fast convergence of BGP routes.

## BGP GR

If BGP restarts, the peer relationship needs to be re-established and traffic forwarding is interrupted. After Graceful Restart (GR) is enabled, traffic interruption is avoided.

## BGP Security

- The AR150/200 authenticates BGP peers by using MD5 and Key-Chain, preventing packet fraud or unauthorized packet modification.
- Generalized TTL Security Mechanism (GTSM) checks TTL values to defend against attacks. GTSM checks whether or not the TTL value in the IP header is within a specified range, protecting the router against attacks and improving system security.

- The number of routes received from the BGP peer is limited to prevent the resources from exhausting. See [8.6.3 Configuring to Control the Acceptment of BGP Routing Information](#).
- The lengths of AS paths on the inbound interface and the outbound interface are limited. The excess packets are discarded. See [8.4.7 Configuring AS\\_Path Attributes for Routes](#).

## 8.3 Configuring Basic BGP Functions

Configuring basic BGP functions is the prerequisite to building a BGP network.

### 8.3.1 Establishing the Configuration Task

Basic BGP functions must be configured first when you build up a BGP network.

#### Applicable Environment

BGP can be configured on a network to implement communication among ASs. This section describes how to configure basic BGP functions.

Because BGP uses TCP connections, you need to specify the IP address of the peer when configuring BGP. The BGP peer may not be the neighboring router. The BGP peer relationship can also be established by using logical links. Loopback interface addresses are usually used to establish BGP connections to enhance the stability of these connections.

Configuring basic BGP functions includes the following steps:

- Start BGP processes. This step is a prerequisite for configuring basic BGP functions.
- Establish BGP peer relationships: Devices can exchange BGP routing information only after they are configured as peers and establish peer relationships.
- Import routes. BGP itself cannot discover routes. Instead, it imports routes discovered by other protocols to implement communication between ASs.

#### NOTE

The commands in the BGP-IPv4 unicast address family view can be run in the BGP view. These commands are described in the BGP-IPv4 unicast address family view in configuration files.

#### Pre-configuration Tasks

Before configuring basic BGP functions, complete the following task:

- Configuring link layer protocol parameters and IP addresses for interfaces to ensure that the link layer protocol on the interfaces is Up

#### Data Preparation

To configure basic BGP functions, you need the following data.

| No. | Data                                 |
|-----|--------------------------------------|
| 1   | Local AS number and router ID        |
| 2   | IPv4 address and AS number of a peer |

| No. | Data                                    |
|-----|-----------------------------------------|
| 3   | Interface originating an Update message |

## 8.3.2 Starting a BGP Process

Starting a BGP process is a prerequisite for configuring basic BGP functions. When starting a BGP process on a device, specify the number of the AS to which the device belongs.

### Context

Do as follows on the router where a BGP connection needs to be established:

### Procedure

**Step 1** Run:

```
system-view
```

The system view is displayed.

**Step 2** Run:

```
bgp as-number
```

BGP is enabled (the local AS number is specified), and the BGP view is displayed.

**Step 3** (Optional) Run:

```
router-id ipv4-address
```

A router ID is set.

Configuring or changing the router ID of BGP causes the BGP peer relationship between routers to be reset.

 **NOTE**

To enhance network reliability, configuring a loopback interface address as the router ID is recommended. If no router ID is set, BGP automatically selects the router ID in the system view as the router ID of BGP. For the rule for selecting a router ID in the system view, see the **router-id** command .

----End

## 8.3.3 Configuring BGP Peers

Two devices can exchange BGP routing information only after they are configured as peers and establish a peer relationship.

### Context

Because BGP uses TCP connections, you need to specify IP addresses for peers when configuring BGP. Two BGP peers are not definitely neighboring to each other. Such BGP peers establish a BGP peer relationship by using a logical link. Using loopback interface addresses to set up BGP peer relationships improves the stability of BGP connections, and therefore is recommended.

IBGP peer relationships are established between the devices within an AS. EBGP peer relationships are established between the devices in different ASs.

## Procedure

- Configure an IBGP peer.

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
bgp as-number
```

The BGP view is displayed.

3. Run:

```
peer ipv4-address as-number as-number
```

The IP address of a peer and the number of the AS where the peer resides are specified.

The number of the AS where the specified peer resides must be the same as that of the local AS.

The IP address of the specified peer can be one of the following types:

- IP address of an interface on a directly-connected peer
- IP address of a loopback interface on a reachable peer
- IP address of a sub-interface on a directly-connected peer

4. Run:

```
peer ipv4-address connect-interface interface-type interface-number
[ipv4-source-address]
```

The source interface and source address are specified for establishing a TCP connection.

By default, BGP uses the physical interface that is directly connected to the peer as the local interface of a TCP connection.

 **NOTE**

When loopback interfaces are used to establish a BGP connection, run the **peer connect-interface** command at both ends of the connection to ensure that the connection is correctly established. If this command is run on only one end, the BGP connection may fail to be established.

5. (Optional) Run:

```
peer ipv4-address description description-text
```

A description is configured for the peer.

Configuring a description for a peer simplifies network management.

- Configure an EBGP peer.

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
bgp as-number
```

The BGP view is displayed.

3. Run:

```
peer ipv4-address as-number as-number
```

The IP address of a peer and the number of the AS where the peer resides are specified.

The number of the AS where the specified peer resides must be different from that of the local AS.

The IP address of the specified peer can be one of the following types:

- IP address of an interface on a directly-connected peer
- IP address of a loopback interface on a reachable peer
- IP address of a sub-interface on a directly-connected peer

4. (Optional) Run:

```
peer ipv4-address connect-interface interface-type interface-number
[ipv4-source-address]
```

The source interface and source address are specified for establishing a TCP connection.

By default, BGP uses the physical interface that is directly connected to the peer as the local interface of a TCP connection.

 **NOTE**

When loopback interfaces are used to establish a BGP connection, run the **peer connect-interface** command at both ends of the connection to ensure that the connection is correctly established. If this command is run on only one end, the BGP connection may fail to be established.

5. (Optional) Run:

```
peer ipv4-address ebgp-max-hop [hop-count]
```

The default value of *hop-count* is 255.

The maximum number of hops is configured for establishing an EBGP connection.

A direct physical link must be available between EBGP peers. If such a link does not exist, the **peer ebgp-max-hop** command must be used to allow EBGP peers to establish a TCP connection over multiple hops.

 **NOTE**

If loopback interfaces are used to establish an EBGP peer relationship, the **peer ebgp-max-hop** command (*hop-count* ≥ 2) must be run. Otherwise, the peer relationship cannot be established.

6. (Optional) Run:

```
peer ipv4-address description description-text
```

A description is configured for the peer.

Configuring a description for a peer simplifies network management.

----End

## 8.3.4 Configuring BGP to Import Routes

BGP can import routes from other protocols. When routes are imported from a dynamic routing protocol, the process IDs of the routing protocol must be specified.




## Context

BGP itself cannot discover routes. Instead, it imports routes discovered by other protocols such as an IGP or the static routing protocol into the BGP routing table. These imported routes are then transmitted within an AS or between ASs.

BGP can import routes in either Import or Network mode:

- In Import mode, BGP imports routes by a specific routing protocol. RIP routes, OSPF routes, IS-IS routes, static routes, or direct routes can be imported into the BGP routing table.
- In Network mode, routes with the specified prefix and mask are imported into the BGP routing table. Compared with the Import mode, the Network mode imports more specific routes.

## Procedure

- Configure BGP to import routes in Import mode.
  1. Run:  
`system-view`  
The system view is displayed.
  2. Run:  
`bgp as-number`  
The BGP view is displayed.
  3. (Optional) Run:  
`ipv4-family unicast`  
The BGP-IPv4 unicast address family view is displayed.  
By default, the BGP-IPv4 unicast address family view is displayed.
  4. Run:  
`import-route protocol [ process-id ] [ med med | route-policy route-policy-name ] *`  
BGP is configured to import routes from other protocols.  
By configuring the parameter `med`, you can set MED values for the imported routes. The EBGP peer selects the route with the smallest MED for traffic entering an AS.  
By configuring the parameter `route-policy route-policy-name`, you can filter the routes imported from other protocols.  
 **NOTE**  
The process ID of a routing protocol needs to be specified if IS-IS, OSPF, or RIP routes are to be imported.
  5. (Optional) Run:  
`default-route imported`  
BGP is configured to import default routes.  
To import default routes, run both the `default-route imported` command and the `import-route` command. If only the `import-route` command is used, no default route

can be imported. In addition, the **default-route imported** command is used to import only the default routes that exist in the local routing table.

- Configure BGP to import routes in Network mode.

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
bgp as-number
```

The BGP view is displayed.

3. (Optional) Run:

```
ipv4-family unicast
```

The BGP-IPv4 unicast address family view is displayed.

By default, the BGP-IPv4 unicast address family view is displayed.

4. Run:

```
network ipv4-address [mask | mask-length] [route-policy route-policy-name]
```

BGP is configured to advertise local routes.

If no mask or mask length is specified, the IP address is processed as a classful address.

A local route to be advertised must be in the local IP routing table. Routing policies can be used to control the routes to be advertised more flexibly.

#### NOTE

- The destination address and mask specified in the **network** command must be consistent with those of the corresponding entry in the local IP routing table. Otherwise, the specified route cannot be advertised.
- When using the **undo network** command to clear the existing configuration, specify a correct mask.

----End

## 8.3.5 Checking the Configuration

After basic BGP functions are configured, you can view information about BGP peers and BGP routes.

### Prerequisites

The configurations of basic BGP functions are complete.

### Procedure

- Run the **display bgp peer [ verbose ]** command to check information about all BGP peers.
- Run the **display bgp peer ipv4-address { log-info | verbose }** command to check log information of a specified BGP peer.
- Run the **display bgp routing-table [ ipv4-address [ mask | mask-length ] ]** command to check BGP routes.

----End

## 8.4 Configuring BGP Route Attributes

BGP has many route attributes. Configuring route attributes can change route selection results.

### 8.4.1 Establishing the Configuration Task

Before configuring BGP route attributes, familiarize yourself with the applicable environment, complete the pre-configuration tasks, and obtain the data required for the configuration. This will help you complete the configuration task quickly and accurately.

#### Applicable Environment

BGP has many route attributes. You can change route selection results by configuring attributes for routes. Route attributes are listed as follows:

- BGP preference  
Setting the BGP preference can affect route selection between BGP routes and other routing protocols' routes.
- Preferred values  
After preferred values are set for BGP routes, the route with the greatest value is preferred when multiple routes to the same destination exist in the BGP routing table.
- Local\_Pref  
The Local\_Pref attribute has the same function as the preferred value of a route. If both of them are configured for a BGP route, the preferred value takes precedence over the Local\_Pref attribute.
- MED  
The MED attribute is used to determine the optimal route for traffic that enters an AS. The route with the smallest MED value is selected as the optimal route if the other attributes of the routes are the same.
- Next\_Hop  
BGP route selection can be flexibly controlled by changing Next\_Hop attributes for routes.
- AS\_Path  
The AS\_Path attribute is used to prevent routing loops and control route selection.

#### Pre-configuration Tasks

Before configuring BGP route attributes, complete the following tasks:

- Configuring IP addresses for interfaces to ensure IP connectivity between neighboring nodes
- [Configuring Basic BGP Functions](#)

#### Data Preparation

To configure BGP route attributes, you need the following data.

| No. | Data                 |
|-----|----------------------|
| 1   | AS number            |
| 2   | BGP preference value |
| 3   | Local_Pref value     |
| 4   | MED value            |

## 8.4.2 Configuring the BGP Preference

Setting the BGP preference can affect route selection between BGP routes and other routing protocols' routes.

### Context

Multiple dynamic routing protocols can be run on a device at the same time. In this case, there is a problem of route sharing and selecting among routing protocols. To address this problem, the system sets a default preference for each routing protocol. If different protocols have routes to the same destination, the protocol with the highest preference is selected to forward IP packets.

Perform the following steps on a device running BGP.

### Procedure

**Step 1** Run:

```
system-view
```

The system view is displayed.

**Step 2** Run:

```
bgp as-number
```

The BGP view is displayed.

**Step 3** Run:

```
ipv4-family unicast
```

The IPv4 unicast address family view is displayed.

**Step 4** Run:

```
preference { external internal local | route-policy route-policy-name }
```

The BGP preference is set.

The smaller the preference value, the higher the preference.

BGP has the following types of routes:

- EBGp routes learned from peers in other ASs
- IBGP routes learned from peers in the same AS
- Locally originated routes (A locally originated route is a route summarized by using the **summary automatic** command or the **aggregate** command.)

Different preference values can be set for these three types of routes.

In addition, a routing policy can also be used to set the preferences for the routes that match the policy. The routes that do not match the policy use the default preference.

 **NOTE**

At present, the **peer route-policy** command cannot be used to set the BGP preference.

----End

## 8.4.3 Configuring Preferred Values for BGP Routes

After preferred values are set for BGP routes, the route with the greatest value is preferred when multiple routes to the same destination exist in the BGP routing table.

### Procedure

**Step 1** Run:

```
system-view
```

The system view is displayed.

**Step 2** Run:

```
bgp as-number
```

The BGP view is displayed.

**Step 3** Run:

```
peer { group-name | ipv4-address } preferred-value value
```

A preferred value is set for all the routes learned from a specified peer.

The original preferred value of a route learned from a peer defaults to 0.

If there are multiple routes to the same address prefix, the route with the highest preferred value is preferred.

----End

## 8.4.4 Configuring a Default Local\_Pref Attribute for a Device

The Local\_Pref attribute is used to determine the optimal route for traffic that leaves an AS.

### Context

The Local\_Pref attribute is used to determine the optimal route for traffic that leaves an AS. If a BGP device obtains multiple routes from different IBGP peers and these routes have different next hops to the same destination, the BGP device will select the route with the greatest Local\_Pref value.

### Procedure

**Step 1** Run:

```
system-view
```

The system view is displayed.

**Step 2** Run:

```
bgp as-number
```

The BGP view is displayed.

**Step 3** Run:

```
ipv4-family unicast
```

The IPv4 unicast address family view is displayed.

**Step 4** Run:

```
default local-preference preference
```

A default Local\_Pref attribute is set for the local device.

---End

## 8.4.5 Configuring MED Attributes for BGP Routes

The MED attribute equals a metric used in an IGP. The MED attribute is used to determine the optimal route for traffic that enters an AS. The route with the smallest MED value is selected as the optimal route if the other attributes of the routes are the same.

### Context

The MED attribute equals a metric used in an IGP, and is used to determine the optimal route for traffic that enters an AS. If a BGP device obtains multiple routes from different EBGP peers and these routes have different next hops to the same destination, the BGP device will select the route with the smallest MED value.

### Procedure

- Set the default MED value on a device.

Perform the following steps on a BGP device:

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
bgp as-number
```

The BGP view is displayed.

3. Run:

```
ipv4-family unicast
```

The IPv4 unicast address family view is displayed.

4. Run:

```
default med med
```

The default MED value is set.

 **NOTE**

The **default med** command is valid only for routes imported using the **import-route** command and BGP summarized routes on the local device.

- Compare the MED values of the routes from different ASs.

Perform the following steps on a BGP device:

1. Run:  
**system-view**  
The system view is displayed.
2. Run:  
**bgp** *as-number*  
The BGP view is displayed.
3. Run:  
**ipv4-family unicast**  
The IPv4 unicast address family view is displayed.
4. Run:  
**compare-different-as-med**  
The MED values of routes from different ASs are compared.

By default, the BGP device compares the MED values of only routes from different peers in the same AS. This command enables the BGP device to compare the MED values of routes from different ASs.

- Configure the deterministic-MED function.

Perform the following steps on a BGP device:

1. Run:  
**system-view**  
The system view is displayed.
2. Run:  
**bgp** *as-number*  
The BGP view is displayed.
3. Run:  
**ipv4-family unicast**  
The IPv4 unicast address family view is displayed.
4. Run:  
**deterministic-med**  
The deterministic-MED function is enabled.

If the deterministic-MED function is not enabled and an optimal route is to be selected among routes that are received from different ASs and carry the same prefix, the sequence in which routes are received is relevant to the route selection result. After the deterministic-MED function is enabled and an optimal route is to be selected among routes that are received from different ASs and carry the same prefix, routes are first grouped based on the leftmost AS number in the AS\_Path attribute. Routes with the same leftmost AS number are grouped together and compared, and an optimal route is selected in the group. The optimal route in this group is then compared with the optimal routes from other groups to determine the final optimal route. This route selection mode allows the route selection result to be independent of the sequence in which routes are received.

- Configure the method used by BGP to handle the situation where a route has no MED attribute during route selection.

Perform the following steps on a BGP device:

1. Run:  
`system-view`  
The system view is displayed.
2. Run:  
`bgp as-number`  
The BGP view is displayed.
3. Run:  
`ipv4-family unicast`  
The IPv4 unicast address family view is displayed.
4. Run:  
`bestroute med-none-as-maximum`  
The system treats a BGP route as one with the maximum MED value if the route has no MED value.  
  
After the `bestroute med-none-as-maximum` command is run, BGP treats a BGP route as one with the maximum MED value if the route that has no MED attribute when selecting an optimal route. If this command is not run, BGP uses 0 as the MED value for a route that has no MED value.

- Compare the MED values of routes in a confederation.

Perform the following steps on a BGP device:

1. Run:  
`system-view`  
The system view is displayed.
2. Run:  
`bgp as-number`  
The BGP view is displayed.
3. Run:  
`ipv4-family unicast`  
The IPv4 unicast address family view is displayed.
4. Run:  
`bestroute med-confederation`  
The MED values of routes in a confederation are compared.

---End

## 8.4.6 Configuring Next\_Hop Attributes for Routes

Setting Next\_Hop attributes for routes flexibly controls BGP route selection.

### Procedure

- Configure a device to change the next-hop address of a route when the device advertises the route to an IBGP peer.

By default, a device does not change the next-hop address of a route learned from an EBGP peer before forwarding the route to IBGP peers. The next-hop address of a route advertised



by an EBGP peer to this device is the address of the EBGP peer. After being forwarded to IBGP peers, this route cannot become an active route because the next hop is unreachable. The relevant ASBR must be configured to change the next-hop address of the route to the ASBR's own IP address before the ASBR advertises the route to an IBGP peer. The route is active on the IBGP peer if the next hop is reachable.

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
bgp as-number
```

The BGP view is displayed.

3. Run:

```
ipv4-family unicast
```

The IPv4 unicast address family view is displayed.

4. Run:

```
peer { ipv4-address | group-name } next-hop-local
```

The device is configured to change the next-hop address of a route to the device's own IP address before the device advertises the route to an IBGP peer.

By default, a device does not change the next-hop address of a route when advertising the route to an IBGP peer.

 **NOTE**

If BGP load balancing is configured, the local router changes the next-hop address of a route to its own IP address when advertising the route to IBGP peers or peer groups, regardless of whether the **peer next-hop-local** command is used.

- Prevent a device from changing the next-hop address of a route imported from an IGP when the device advertises the route to an IBGP peer.

Perform the following steps on a router that runs BGP and has imported IGP routes:

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
bgp as-number
```

The BGP view is displayed.

3. Run:

```
ipv4-family unicast
```

The IPv4 unicast address family view is displayed.

4. Run:

```
peer { ipv4-address | group-name } next-hop-invariable
```

The device is prevented from changing the next-hop address of a route imported from an IGP before advertising the route to an IBGP peer.

By default, a device changes the next-hop address of a route imported from an IGP to the address of the interface connecting the device to its peer when advertising the route to an IBGP peer.

- Prevent an ASBR from changing the next-hop address of a route when the ASBR advertises the route to an EBGP peer.

Perform the following steps on a BGP device that functions as an ASBR:

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
bgp as-number
```

The BGP view is displayed.

3. Run:

```
ipv4-family vpn4 [unicast]
```

The BGP-VPNv4 sub-address family view is displayed.

4. Run:

```
peer { group-name | ipv4-address } next-hop-invariable
```

The device is prevented from changing the next-hop address of a route when advertising the route to an EBGP peer.

By default, PEs in different ASs set up EBGP peer relationships with each other, and they do not change next-hop addresses of routes when advertising the routes to their EBGP peers.

In the inter-AS VPN option C networking where RRs are used, the **peer next-hop-invariable** command needs to be run to prevent the RRs from changing the next-hop address of a route when the RRs advertise the route to EBGP peers. This ensures that the remote PE iterates a route to the BGP LSP destined for the local PE during traffic transmission.

- Configure routing-policy-based next hop iteration.

Perform the following steps on a BGP device:

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
bgp as-number
```

The BGP view is displayed.

3. Run:

```
ipv4-family unicast
```

The IPv4 unicast address family view is displayed.

4. Run:

```
nexthop recursive-lookup route-policy route-policy-name
```

Routing-policy-based next hop iteration is configured.

By default, routing-policy-based next hop iteration is not configured.

Next-hop iteration based on a specified routing policy can control the iterated next hop based on specific conditions. If a route cannot match the specified routing policy, the route cannot be iterated.

---End

## 8.4.7 Configuring AS\_Path Attributes for Routes

The AS\_Path attribute is used to prevent routing loops and control route selection.

### Procedure

- Allow repeated local AS numbers.

BGP uses AS numbers to detect routing loops. In Hub and Spoke networking, if EBGp runs between a Hub-PE and a Hub-CE, the route sent from the Hub-PE to the Hub-CE carries the AS number of the Hub-PE. After the Hub-CE sends an Update message that contains the AS number of the Hub-PE to the Hub-PE, the Hub-PE will deny it.

To ensure proper route transmission in Hub and Spoke networking, configure all the BGP peers on the path, along which the Hub-CE advertises private network routes to the Spoke-CE, to accept the routes in which the local AS number repeats once.

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
bgp as-number
```

The BGP view is displayed.

3. Run:

```
ipv4-family unicast
```

The IPv4 unicast address family view is displayed.

4. Run:

```
peer { ipv4-address | group-name } allow-as-loop [number]
```

The local AS number is allowed to repeat in the AS\_Path attribute.

Generally, a BGP device checks the AS\_Path attribute of a route sent from a peer. If the local AS number already exists in the AS\_Path attribute, BGP ignores this route to avoid a routing loop.

In some special applications, you can use the **peer allow-as-loop** command to allow the AS\_Path attributes of routes sent from the peers to contain the local AS number. You can also set the number of times the local AS number is repeated.

- Configure BGP not to compare AS\_Path attributes of routes in the route selection process.

Perform the following steps on a BGP device:

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
bgp as-number
```

The BGP view is displayed.

3. Run:  
`ipv4-family unicast`

The IPv4 unicast address family view is displayed.

4. Run:  
`bestroute as-path-ignore`

BGP is configured to ignore AS\_Path attributes of routes during route selection.

- Configure a fake AS number.

Generally, a device supports only one BGP process. This means that a device supports only one AS number. If AS numbers need to be replaced during network migration, you can run the **peer fake-as** command to set a fake AS number for a specified peer to ensure smooth network migration.

1. Run:  
`system-view`

The system view is displayed.

2. Run:  
`bgp as-number`

The BGP view is displayed.

3. Run:  
`peer { ipv4-address | group-name } fake-as fake-as-number`

A fake AS number is configured.

The **peer fake-as** command can be used to hide the actual AS number of a BGP device. EBGP peers in other ASs will use the fake AS number of this BGP device to set up EBGP peer relationships with this device.

 **NOTE**

This command can be used only on EBGP peers.

- Enable AS number replacement.

Before advertising a route to a specified CE, a PE enabled with AS number replacement replaces the AS number of the CE in the AS\_Path attribute of the route with the local AS number.



**CAUTION**

Exercise caution when running the **peer substitute-as** command, because improper use of this command may cause routing loops.

- 
1. Run:  
`system-view`  
The system view is displayed.
  2. Run:  
`bgp as-number`

The BGP view is displayed.

3. Run:

```
ipv4-family vpn-instance vpn-instance-name
```

The BGP-VPN instance IPv4 address family view is displayed.

4. Run:

```
peer { ipv4-address | group-name } substitute-as
```

AS number replacement is enabled.

- Configure the AS\_Path attribute to carry only public AS numbers.

A route advertised by a BGP device to its peer usually carries an AS number. The AS number may be public or private. Public AS numbers can be used on the Internet. They are assigned and managed by the Internet Assigned Number Authority (IANA). Private AS numbers cannot be advertised to the Internet, and they are used only within ASs. If private AS numbers are advertised to the Internet, a routing loop may occur. To address this problem, you can run the **peer public-as-only** command to allow the AS\_Path attribute to carry only public AS numbers.

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
bgp as-number
```

The BGP view is displayed.

3. Run:

```
ipv4-family unicast
```

The IPv4 unicast address family view is displayed.

4. Run:

```
peer { ipv4-address | group-name } public-as-only
```

The AS\_Path attribute is configured to carry only public AS numbers.

An AS number ranges from 1 to 4294967295. A public AS number ranges from 1 to 64511, and from 65536 (1.0 in the x.y format) to 4294967295 (65535.65535 in the x.y format). A private AS number ranges from 64512 to 65534. The AS number 65535 is reserved for particular use.

The **peer public-as-only** command can be used only on EBGPeers.

- Set the maximum number of AS numbers in the AS\_Path attribute.

Perform the following steps on a BGP device:

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
bgp as-number
```

The BGP view is displayed.

3. Run:

```
as-path-limit as-path-limit-num
```

The maximum number of AS numbers in the AS\_Path attribute is set.

By default, a maximum of 255 AS numbers can be contained in the AS\_Path attribute.

After the **as-path-limit** command is run on a device, the device checks whether the number of AS numbers in the AS-Path attribute of a received route exceeds the maximum value. If the number of AS numbers exceeds the maximum value, the route is discarded. If the maximum number of AS numbers in the AS-Path attribute is too small, routes whose number of AS numbers exceeding the maximum value will be discarded.

- Prevent a BGP device from checking the first AS number contained in the AS\_Path attribute of an Update message received from an EBGP peer.

Perform the following steps on a BGP device:

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
bgp as-number
```

The BGP view is displayed.

3. Run:

```
undo check-first-as
```

The BGP device is prevented from checking the first AS number contained in the AS\_Path attribute of an Update message received from an EBGP peer.

By default, a BGP device checks whether the first AS number contained in the AS\_Path attribute of an Update message received from an EBGP peer is the same as the number of the AS where the EBGP peer resides. If the numbers are not the same, the BGP device discards the Update message and closes the EBGP connection with the EBGP peer.



## CAUTION

Exercise caution when running the **undo check-first-as** command, because use of this command increases the possibility of routing loops.

---

After the configuration is complete, run the **refresh bgp** command if you want to check the received routes again.

---End

## 8.4.8 Checking the Configuration

After BGP route attributes are configured, you can view information about these route attributes.

### Prerequisites

The BGP route attribute configuration is complete.

## Procedure

- Run the **display bgp paths** [ *as-regular-expression* ] command to check information about AS\_Path attributes of routes.
- Run the **display bgp routing-table different-origin-as** command to check information about routes that have the same destination address but different source AS numbers.
- Run the **display bgp routing-table regular-expression** *as-regular-expression* command to check information about routes matching a specified regular expression.
- Run the **display bgp routing-table** [ *network* [ { *mask* | *mask-length* } ] [ **longer-prefixes** ] ] command to check routing information in a BGP routing table.

----End

## 8.5 Configuring BGP to Advertise Routes

BGP is used to transmit routing information. BGP advertises only the wanted routes after filtering routes to be advertised, and modifies route attributes to direct network traffic.

### 8.5.1 Establishing the Configuration Task

Before configuring BGP to advertise routes, familiarize yourself with the applicable environment, complete the pre-configuration tasks, and obtain the data required for the configuration. This will help you complete the configuration task quickly and efficiently.

#### Applicable Environment

BGP is used to transmit routing information between ASs. Route advertisement directly affects traffic forwarding.

There are usually a large number of routes in a BGP routing table. Transmitting a great deal of routing information brings a heavy load to devices. Routes to be advertised need to be controlled to address this problem. You can configure devices to advertise only routes that these devices want to advertise or routes that their peers require.

Multiple routes to the same destination may exist and traverse different ASs. Routes to be advertised need to be filtered in order to direct routes to specific ASs.

Filters can be used to filter routes to be advertised by BGP. BGP can filter routes to be advertised to a specific peer or peer group.

#### Pre-configuration Tasks

Before configuring BGP to advertise routes, complete the following task:

- **Configuring Basic BGP Functions**

#### Data Preparation

To configure BGP to advertise routes, you need the following data.

| No. | Data                                                                            |
|-----|---------------------------------------------------------------------------------|
| 1   | Name or number of an ACL                                                        |
| 2   | Name, number, and matching mode of an IP prefix list                            |
| 3   | Number or name of an AS_Path filter                                             |
| 4   | Number or name and matching mode of a community filter                          |
| 5   | Number or name and matching mode of an extcommunity filter                      |
| 6   | Name and matching mode of a route-policy, and number of the route-policy's node |

## 8.5.2 Configuring BGP Filters

BGP filters filter routes to be advertised.

### Context

BGP uses the following types of filters to filter routes:

- [Access Control List\(ACL\)](#)
- [IP-Prefix List](#)
- [AS\\_Path filter](#)
- [Community filter](#)
- [Extcommunity filter](#)
- [Route-Policy](#)

### Procedure

- Configure an ACL.

An ACL is a series of sequential rules composed of **permit** and **deny** clauses. These rules are described based on source addresses, destination addresses, and port numbers of packets. ACL rules are used to classify packets. After ACL rules are applied to a device, the device permits or denies packets based on the ACL rules.

For details on ACL configurations, see the *Huawei AR150&200 Series Enterprise Routers Configuration Guide - IP Services*.

An ACL can be used as a matching condition of a route-policy or used in the **filter-policy** { *acl-number* | **acl-name** *acl-name* } **export** [ *protocol* [ *process-id* ] ] command or the **peer** { *group-name* | *ipv4-address* } **filter-policy** { *acl-number* | **acl-name** *acl-name* } **export** command.

- Configure an IP prefix list.

An IP prefix list is a type of filter used to filter routes based on destination addresses. An IP prefix list is identified by its name. An IP prefix list can be used flexibly to implement accurate filtering. For example, it can be used to filter a route or routes to a network segment. If a large number of routes that do not have the same prefix need to be filtered, configuring an IP prefix list to filter the routes is very complex.



An IP prefix list can be used as a matching condition of a route-policy or used in the **filter-policy ip-prefix** *ip-prefix-name* **export** [ *protocol* [ *process-id* ] ] command or the **peer** { *group-name* | *ipv4-address* } **ip-prefix** *ip-prefix-name* **export** command.

Perform the following steps on a BGP router:

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
ip ip-prefix ip-prefix-name [index index-number] { permit | deny } ip-
address mask-length [greater-equal greater-equal-value] [less-equal
less-equal-value]
```

An IPv4 prefix list is configured.

The mask length range can be specified as *mask-length* <= *greater-equal-value* <= *less-equal-value* <= 32. If only **greater-equal** is specified, the prefix range is [*greater-equal-value*, 32]. If only **less-equal** is specified, the prefix range is [*mask-length*, *less-equal-value*].

An IPv4 prefix list is identified by its name, and each IP prefix list can contain multiple entries. Each entry is identified by an index number, and can specify a matching range in the form of a network prefix uniquely. An IPv4 prefix list named **abcd** is used as an example.

```

ip ip-prefix abcd index 10 permit 1.0.0.0 8
ip ip-prefix abcd index 20 permit 2.0.0.0 8
```

During route matching, the system checks the entries by index number in ascending order. If a route matches an entry, the route will not be matched with the next entry.

The AR150/200 denies all unmatched routes by default. If all entries in an IPv4 prefix list are in deny mode, all routes will be denied by the IPv4 prefix list. In this case, you must define an entry **permit 0.0.0.0 0 less-equal 32** after the entries in deny mode to allow all the other IPv4 routes to be permitted by the IPv4 prefix list.

#### NOTE

If more than one IP prefix entry is defined, at least one entry should be set in permit mode.

- Configure an AS\_Path filter.

An AS\_Path filter is used to filter BGP routes based on the AS\_Path attributes contained in the BGP routes. If you do not want traffic to pass through an AS, configure an AS\_Path filter to filter out the traffic carrying the number of the AS. If the BGP routing table of each device on a network is large, configuring an ACL or an IP prefix list to filter BGP routes may be complicated and make it difficult to maintain new routes.

#### NOTE

If the AS\_Path information of a summarized route is lost, the AS\_Path filter cannot be used to filter the summarized route, but can still be used to filter the specific routes from which the summarized route is derived.

An AS\_Path filter can be used as a matching condition of a route-policy or be used in the **peer as-path-filter** command.

Perform the following steps on a BGP router:

## 1. Run:

```
system-view
```

The system view is displayed.

## 2. Run:

```
ip as-path-filter { as-path-filter-number | as-path-filter-name }
{ permit | deny } regular-expression
```

An AS\_Path filter is configured.

An AS\_Path filter uses a regular expression to define matching rules. A regular expression consists of the following parts:

- Metacharacter: defines matching rules.
- General character: defines matching objects.

**Table 8-1** Metacharacters

| Metacharacter | Description                                                                                                                                                                                         |
|---------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| \             | Escape character.                                                                                                                                                                                   |
| .             | Matches any single character except "\n", including spaces.                                                                                                                                         |
| *             | An asterisk indicates that there are 0, 1, or any number of the previous expression.                                                                                                                |
| +             | A plus sign indicates that there is at least 1 of the previous expression.                                                                                                                          |
|               | Matches either expression it separates.                                                                                                                                                             |
| ^             | Specifies the beginning of a line.                                                                                                                                                                  |
| \$            | Specifies the end of a line.                                                                                                                                                                        |
| [xyz]         | Matches any character in the brackets.                                                                                                                                                              |
| [^xyz]        | Matches a single character that is not contained within the brackets.                                                                                                                               |
| [a-z]         | Matches any character within the specified range.                                                                                                                                                   |
| [^a-z]        | Matches any character out of the specified range.                                                                                                                                                   |
| {n}           | Repeats "n" times. "n" is a non-negative integer.                                                                                                                                                   |
| {n,}          | Repeats at least "n" times. "n" is a non-negative integer.                                                                                                                                          |
| {n,m}         | Repeats "n" to "m" times. "m" and "n" are both non-negative integers, and "n" is equal to or smaller than "m". Note that there is no space between "n" and the comma, or between the comma and "m". |

For example, ^10 indicates that only the AS\_Path attribute starting with 10 is matched. A circumflex (^) indicates that the beginning of a character string is matched.

Multiple rules, permit or deny, can be specified in a filter. The relationship between these rules is "OR". This means that if a route meets one of the matching rules, the route matches the AS\_Path filter.

 **NOTE**

For details on a regular expression, see the *Huawei AR150&200 Series Enterprise Routers Configuration Guide - Basic Configurations*.

- Configure a community filter.

A BGP community attribute is used to identify a group of routes with the same properties. Routes can be classified by community attribute. This facilitates route management.

Some AS internal routes may not need to be advertised to any other AS, whereas AS external routes need to be advertised to other ASs. These AS external routes have different prefixes (as a result, an IP prefix list is inapplicable) and may come from different ASs (as a result, an AS\_Path filter is inapplicable). You can set a community attribute value for these AS internal routes and another community attribute value for these AS external routes on an ASBR to control and filter these routes.

Perform the following steps on a BGP router:

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
ip community-filter
```

A community filter is configured.

- To configure a standard community filter, run the **ip community-filter { basic comm-filter-name { permit | deny } [ community-number | aa:nn ] \* <1-9> | basic-comm-filter-num { permit | deny } [ community-number | aa:nn ] \* <1-16> } [ internet | no-export-subconfed | no-advertise | no-export ] \*** command.
- To configure an advanced community filter, run the **ip community-filter { advanced comm-filter-name | adv-comm-filter-num } { permit | deny } regular-expression** command.

- Configure an extcommunity filter.

Similar to a BGP community filter, a BGP extcommunity filter is used to filter private network routes.

Perform the following steps on a BGP router:

1. Run:

```
system-view
```

The system view is displayed.

2. Perform either of the following operations as required to configure an extcommunity filter.

- To configure a basic extcommunity filter, run the **ip extcommunity-filter { basic-extcomm-filter-num | basic basic-extcomm-filter-name } { deny | permit } { rt { as-number:nn | ipv4-address:nn } } <1-16>** command.

- To configure an advanced extcommunity filter, run the **ip extcommunity-filter** { *adv-extcomm-filter-num* | **advanced** *adv-extcomm-filter-name* } { **deny** | **permit** } *regular-expression* command.

Multiple entries can be defined in an extcommunity filter. The relationship between the entries is "OR". This means that if a route matches one of the rules, the route matches the filter.

- Configure a route-policy.

A route-policy is used to match routes or route attributes, and to change route attributes when specific conditions are met. As the preceding filters can be used as matching conditions of a route-policy, the route-policy is powerful in functions and can be used flexibly.

Perform the following steps on a BGP router:

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
route-policy route-policy-name { permit | deny } node node
```

A node is configured for a route-policy, and the view of the route-policy is displayed.

A route-policy consists of multiple nodes. For example, the **route-policy route-policy-example permit node 10** command specifies node 10 and the **route-policy route-policy-example deny node 20** command specifies node 20. The two nodes belong to the route-policy specified by **route-policy-example**. The relationship between the nodes of a route-policy is "OR". The details are as follows:

- If a route matches one node, the route matches the route-policy and will not be matched with the next node. For example, there are two nodes defined using the **route-policy route-policy-example permit node 10** and **route-policy route-policy-example deny node 20** commands. If a route matches the node defined using the **route-policy route-policy-example permit node 10** command, the route will not be matched with the node defined using the **route-policy route-policy-example deny node 20** command.
- If a route does not match any node, the route fails to match the route-policy.

When a route-policy is used to filter a route, the route is first matched with the node with the smallest *node* value. For example, if two nodes are configured using the **route-policy route-policy-example permit node 10** and **route-policy route-policy-example deny node 20** commands, a route is first matched with the node configured using the **route-policy route-policy-example permit node 10** command.

 **NOTE**

The AR150/200 considers that each unmatched route fails to match the route-policy by default. If more than one node is defined in a route-policy, at least one of them must be in **permit** mode.

3. (Optional) Perform the following operations as needed to configure **if-match** clauses for current nodes of the route-policy.

**if-match** clauses are used to filter routes. If no **if-match** clause is specified, all routes will match the node in the route-policy.

- To match an ACL, run the **if-match acl** { *acl-number* | *acl-name* } command.

- To match an IP prefix list, run the **if-match ip-prefix** *ip-prefix-name* command.

 **NOTE**

The **if-match acl** and **if-match ip-prefix** commands cannot be used together in the same node of a route-policy, because the latest configuration will override the previous one.

- To match the AS-Path attribute of BGP routes, run the **if-match as-path-filter** { *as-path-filter-number* | *as-path-filter-name* } &<1-16> command.
- To match the community attribute of BGP routes, run either of the following commands:
  - **if-match community-filter** { *basic-comm-filter-num* [ **whole-match** ] | *adv-comm-filter-num* } \* &<1-16>
  - **if-match community-filter** *comm-filter-name* [ **whole-match** ]
- To match the extended community attribute of BGP routes, run the **if-match extcommunity-filter** { { *basic-extcomm-filter-num* | *adv-extcomm-filter-num* } &<1-16> | *basic-extcomm-filter-name* | *advanced-extcomm-filter-name* } command.

The operations in Step 3 can be performed in any order. A node may have multiple **if-match** clauses or no **if-match** clause.

 **NOTE**

The relationship between the **if-match** clauses in a node of a route-policy is "AND". A route must match all the rules before the action defined by the **apply** clause is taken. For example, if two **if-match** clauses (**if-match acl 2003** and **if-match as-path-filter 100**) are defined in the **route-policy route-policy-example permit node 10** command, a route is considered to match node 10 only when it matches the two **if-match** clauses.

4. (Optional) Perform the following operations as needed to configure **apply** clauses for current nodes of the route-policy:

**apply** clauses can be used to set attributes for routes matching **if-match** clauses. If this step is not performed, the attributes of routes matching **if-match** clauses keep unchanged.

- To replace or add a specified AS number in the AS\_Path attribute of a BGP route, run the **apply as-path** *as-number* command.
- To delete a specified BGP community attribute from a route, run the **apply community-filter** *comm-filter-number delete* command.

 **TIP**

The **apply community-filter delete** command deletes a specified community attribute from a route. An instance of the **ip community-filter** command can specify only one community attribute each time. To delete more than one community attribute, run the **ip community-filter** command multiple times. If multiple community attributes are specified in one community filter, none of them can be deleted. For more information, see the *Huawei AR150&200 Series Enterprise Routers Command Reference*.

- To delete all community attributes from a BGP route, run the **apply community none** command.
- To set community attributes for a BGP route, run the **apply community** { { *community-number* | *aa:nn* } &<1-32> | **internet** | **no-advertise** | **no-export** | **no-export-subconfed** } \* [ **additive** ] command.
- To set an extended community attribute (route-target) for a route, run the **apply extcommunity** { **rt** { *as-number:nn* | *4as-number:nn* | *ipv4-address:nn* } } &<1-16> [ **additive** ] command.

- To set the local preference for a BGP route, run the **apply local-preference preference** command.
- To set the Origin attribute for a BGP route, run the **apply origin { igp | egp as-number | incomplete }** command.
- To set a preferred value for a BGP route, run the **apply preferred-value preferred-value** command.
- To set dampening parameters for an EBGP route, run the **apply dampening half-life-reach reuse suppress ceiling** command.

The operations in Step 4 can be performed in any order. A node may have multiple **apply** clauses or no **apply** clause.

---End

## 8.5.3 Configuring to Control the Advertisement of BGP Routing Information

After a route advertisement policy is configured on a device, the device advertises only routes matching the policy to its peers.

### Procedure

- Configure a BGP device to advertise routes to all peers or peer groups.

You can configure a BGP device to filter routes to be advertised. Perform the following steps on a BGP router:

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
bgp as-number
```

The BGP view is displayed.

3. Run:

```
ipv4-family unicast
```

The IPv4 unicast address family view is displayed.

4. Perform either of the following operations to configure the BGP device to advertise routes to all peers or peer groups:

- To filter routes based on an ACL, run the **filter-policy { acl-number | acl-name acl-name } export [ protocol [ process-id ] ]** command.
- To filter routes based on an IP prefix list, run the **filter-policy ip-prefix ip-prefix-name export [ protocol [ process-id ] ]** command.

If *protocol* is specified, only routes discovered by a specific routing protocol are filtered. If *protocol* is not specified, all the routes to be advertised are filtered, including routes imported using the **import-route (BGP)** command and local routes advertised using the **network (BGP)** command.

 **NOTE**

If an ACL has been referenced in the **filter-policy** command but no VPN instance is specified in the ACL rule, BGP will filter routes including public and private network routes in all address families. If a VPN instance is specified in the ACL rule, only the data traffic from the VPN instance will be filtered, and no route of this VPN instance will be filtered.

- Configure a BGP device to advertise routes to a specific peer or peer group.

You can configure a BGP device to filter routes to be advertised. Perform the following steps on a BGP router:

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
bgp as-number
```

The BGP view is displayed.

3. Run:

```
ipv4-family unicast
```

The IPv4 unicast address family view is displayed.

4. Perform any of the following operations to configure the BGP device to advertise routes to a specific peer or peer group:

- To filter routes based on an ACL, run the **peer** { *ipv4-address* | *group-name* } **filter-policy** { *acl-number* | **acl-name** *acl-name* } **export** command.
- To filter routes based on an IP prefix list, run the **peer** { *ipv4-address* | *group-name* } **ip-prefix** *ip-prefix-name* **export** command.
- To filter routes based on an AS\_Path filter, run the **peer** { *ipv4-address* | *group-name* } **as-path-filter** { *as-path-filter-number* | *as-path-filter-name* } **export** command.
- To filter routes based on a route-policy, run the **peer** { *ipv4-address* | *group-name* } **route-policy** *route-policy-name* **export** command.

A peer group and its members can use different export policies to filter routes. Each peer can select its policy when advertising routes.

----End

## 8.5.4 Configuring BGP Soft Reset

BGP soft reset allows the system to refresh a BGP routing table dynamically without tearing down any BGP connection if routing policies are changed.

### Context

After changing a BGP import policy, you must reset BGP connections for the new import policy to take effect, interrupting these BGP connections temporarily. BGP route-refresh allows the system to refresh a BGP routing table dynamically without tearing down any BGP connection if routing policies are changed.

- If a device's peer supports route-refresh, the **refresh bgp** command can be used on the device to softly reset the BGP connection with the peer and update the BGP routing table.

- If a device's peer does not support route-refresh, the **peer keep-all-routes** command can be used on the device to remain all routing updates received from the peer so that the device can refresh its routing table without closing the connection with the peer.

Perform the following steps on a BGP router:

## Procedure

- If the device's peers support route-refresh, perform the following operations:

1. (Optional) Enable route-refresh.

- a. Run:

```
system-view
```

The system view is displayed.

- b. Run:

```
bgp as-number
```

The BGP view is displayed.

- c. Run:

```
peer { ipv4-address | group-name } capability-advertise route-refresh
```

Route-refresh is enabled.

By default, route-refresh is enabled.

If route-refresh is enabled on all BGP routers and the import policy of the local router is changed, the local router sends a route-refresh message to peers or peer groups. After receiving the message, the peers or peer groups resend routing information to the local BGP router. This enables the local router to dynamically refresh its BGP routing table and apply the new routing policy without closing any BGP connections.

2. Configure BGP soft reset.

- a. Run the **refresh bgp [ vpn-instance vpn-instance-name ipv4-family ] { all | ipv4-address | group group-name | external | internal } { export | import }** command in the user view to softly reset the BGP connections between the devices and its peers or peer groups.

**external** softly resets an EBGP connection, and **internal** softly resets an IBGP connection.

**export** triggers outbound BGP soft reset, and **import** triggers inbound BGP soft reset.

- If the device's peers do not support route-refresh, perform the following operations:

- Configure the device to store all the routing updates received from its peers or peer groups.

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
bgp as-number
```

The BGP view is displayed.

3. Run:

```
ipv4-family unicast
```



The IPv4 unicast address family view is displayed.

4. Run:

```
peer { ipv4-address | group-name } keep-all-routes
```

The device is configured to store all the routing updates received from its peers or peer groups.

By default, the device stores only the routing updates that are received from peers or peer groups and match a configured import policy.

After this command is used, all routing updates sent by a specified peer or peer group are stored, regardless of whether an import policy is used. When the local routing policy changes, the information can be used to regenerate BGP routes again.

 **NOTE**

This command must be run on the local device and its peers. If the **peer keep-all-routes** command is run on the device for the first time, the sessions between the device and its peers are reestablished.

The **peer keep-all-routes** command does not need to be run on the router that supports route-refresh. If the **peer keep-all-routes** command is run on the router, the sessions between the router and its peers will not be reestablished but the **refresh bgp** command does not take effect on the router.

----End

## 8.5.5 Checking the Configuration

After the configurations of controlling BGP route advertisement are complete, you can view filters, routes matching a specified filter, and routes advertised to BGP peers.

### Prerequisites

The BGP route advertisement configurations are complete.

### Procedure

- Run the **display ip as-path-filter** [ *as-path-filter-number* | *as-path-filter-name* ] command to check information about a configured AS\_Path filter.
- Run the **display ip community-filter** [ *basic-comm-filter-num* | *adv-comm-filter-num* | *comm-filter-name* ] command to check information about a configured community filter.
- Run the **display ip extcommunity-filter** [ *extcomm-filter-number* | *extcomm-filter-name* ] command to check information about a configured extcommunity filter.
- Run the **display bgp routing-table as-path-filter** { *as-path-filter-number* | *as-path-filter-name* } command to check information about routes matching a specified AS\_Path filter.
- Run the **display bgp routing-table community-filter** { { *community-filter-name* | *basic-community-filter-number* } [ **whole-match** ] | *advanced-community-filter-number* } command to check information about routes matching a specified BGP community filter.
- Run the **display bgp routing-table peer ipv4-address advertised-routes** [ **statistics** ] command to check information about routes advertised by a BGP device to its peers.

----End

## 8.6 Configuring BGP to Receive Routes

BGP is used to transmit routing information. BGP can filter received routes to accept only the expected routes, and can modify route attributes to direct network traffic.

### 8.6.1 Establishing the Configuration Task

Before configuring BGP to receive routes, familiarize yourself with the applicable environment, complete the pre-configuration tasks, and obtain the data required for the configuration. This will help you complete the configuration task quickly and efficiently.

#### Applicable Environment

BGP is used to transmit routing information between ASs. Route reception directly affects traffic forwarding.

The BGP router may receive routes to the same destination from different BGP peers. To control traffic forwarding paths, the router needs to filter the received BGP routes.

The router may be attacked and receive a large number of routes from its BGP peers, consuming lots of resources of the router. Therefore, the administrator must limit the resources to be consumed based on networking planning and router capacities, no matter whether too many BGP routes caused by malicious attacks or incorrect configurations.

Filters can be used to filter routes to be received by BGP. BGP can filter the routes received from all peers or peer groups or only the routes received from a specific peer or peer group.

#### Pre-configuration Tasks

Before configuring BGP to receive routes, complete the following task:

- [Configuring Basic BGP Functions](#)

#### Data Preparation

To configure BGP to receive routes, you need the following data.

| No. | Data                                                                            |
|-----|---------------------------------------------------------------------------------|
| 1   | Name or number of an ACL                                                        |
| 2   | Name, number, and matching mode of an IP prefix list                            |
| 3   | Number or name of an AS_Path filter                                             |
| 4   | Number or name and matching mode of a community filter                          |
| 5   | Number or name and matching mode of an extended community filter                |
| 6   | Name and matching mode of a route-policy, and number of the route-policy's node |

## 8.6.2 Configuring BGP Filters

BGP filters can be used to filter routes to be received.

### Context

Filters are needed to filter routes to flexibly receive routes. Currently, six filters are available for BGP:

- [Access Control List\(ACL\)](#)
- [IP-Prefix List](#)
- [AS\\_Path filter](#)
- [Community filter](#)
- [Extcommunity filter](#)
- [Route-Policy](#)

### Procedure

- Configure an ACL.

An ACL is a series of sequential rules composed of **permit** and **deny** clauses. These rules are described based on source addresses, destination addresses, and port numbers of packets. ACL rules are used to classify packets. After ACL rules are applied to a device, the device permits or denies packets based on the ACL rules.

For details on ACL configurations, see the *Huawei AR150&200 Series Enterprise Routers Configuration Guide - IP Services*.

An ACL can be used as a matching condition of a route-policy or used in the **filter-policy** { *acl-number* | **acl-name** *acl-name* } **import** command or the **peer** { *group-name* | *ipv4-address* } **filter-policy** { *acl-number* | **acl-name** *acl-name* } **import** command.

- Configure an IP prefix list.

An IP prefix list is a type of filter used to filter routes based on destination addresses. An IP prefix list is identified by its name. An IP prefix list can be used flexibly to implement accurate filtering. For example, it can be used to filter a route or routes to a network segment. If a large number of routes that do not have the same prefix need to be filtered, configuring an IP prefix list to filter the routes is very complex.

An IP prefix list can be used as a matching condition of a route-policy or used in the **filter-policy ip-prefix** *ip-prefix-name* **import** command or the **peer** { *group-name* | *ipv4-address* } **ip-prefix** *ip-prefix-name* **import** command.

Perform the following steps on a BGP router:

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
ip ip-prefix ip-prefix-name [index index-number] { permit | deny } ip-
address mask-length [greater-equal greater-equal-value] [less-equal
less-equal-value]
```

An IPv4 prefix list is configured.

The mask length range can be specified as *mask-length* <= *greater-equal-value* <= *less-equal-value* <= 32. If only **greater-equal** is specified, the prefix range is [*greater-equal-value*, 32]. If only **less-equal** is specified, the prefix range is [*mask-length*, *less-equal-value*].

An IPv4 prefix list is identified by its name, and each IP prefix list can contain multiple entries. Each entry is identified by an index number, and can specify a matching range in the form of a network prefix uniquely. An IPv4 prefix list named **abcd** is used as an example.

```
#
ip ip-prefix abcd index 10 permit 1.0.0.0 8
ip ip-prefix abcd index 20 permit 2.0.0.0 8
```

During route matching, the system checks the entries by index number in ascending order. If a route matches an entry, the route will not be matched with the next entry.

The AR150/200 denies all unmatched routes by default. If all entries in an IPv4 prefix list are in deny mode, all routes will be denied by the IPv4 prefix list. In this case, you must define an entry **permit 0.0.0.0 0 less-equal 32** after the entries in deny mode to allow all the other IPv4 routes to be permitted by the IPv4 prefix list.

#### NOTE

If more than one IP prefix entry is defined, at least one entry should be set in permit mode.

#### ● Configure an AS\_Path filter.

An AS\_Path filter is used to filter BGP routes based on the AS\_Path attributes contained in the BGP routes. If you do not want traffic to pass through an AS, configure an AS\_Path filter to filter out the traffic carrying the number of the AS. If the BGP routing table of each device on a network is large, configuring an ACL or an IP prefix list to filter BGP routes may be complicated and make it difficult to maintain new routes.

#### NOTE

If the AS\_Path information of a summarized route is lost, the AS\_Path filter cannot be used to filter the summarized route, but can still be used to filter the specific routes from which the summarized route is derived.

An AS\_Path filter can be used as a matching condition of a route-policy or be used in the **peer as-path-filter** command.

Perform the following steps on a BGP router:

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
ip as-path-filter { as-path-filter-number | as-path-filter-name }
{ permit | deny } regular-expression
```

An AS\_Path filter is configured.

An AS\_Path filter uses a regular expression to define matching rules. A regular expression consists of the following parts:

- Metacharacter: defines matching rules.
- General character: defines matching objects.

**Table 8-2** Metacharacters

| Metacharacter | Description                                                                                                                                                                                         |
|---------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| \             | Escape character.                                                                                                                                                                                   |
| .             | Matches any single character except "\n", including spaces.                                                                                                                                         |
| *             | An asterisk indicates that there are 0, 1, or any number of the previous expression.                                                                                                                |
| +             | A plus sign indicates that there is at least 1 of the previous expression.                                                                                                                          |
|               | Matches either expression it separates.                                                                                                                                                             |
| ^             | Specifies the beginning of a line.                                                                                                                                                                  |
| \$            | Specifies the end of a line.                                                                                                                                                                        |
| [xyz]         | Matches any character in the brackets.                                                                                                                                                              |
| [^xyz]        | Matches a single character that is not contained within the brackets.                                                                                                                               |
| [a-z]         | Matches any character within the specified range.                                                                                                                                                   |
| [^a-z]        | Matches any character out of the specified range.                                                                                                                                                   |
| {n}           | Repeats "n" times. "n" is a non-negative integer.                                                                                                                                                   |
| {n,}          | Repeats at least "n" times. "n" is a non-negative integer.                                                                                                                                          |
| {n,m}         | Repeats "n" to "m" times. "m" and "n" are both non-negative integers, and "n" is equal to or smaller than "m". Note that there is no space between "n" and the comma, or between the comma and "m". |

For example, ^10 indicates that only the AS\_Path attribute starting with 10 is matched. A circumflex (^) indicates that the beginning of a character string is matched.

Multiple rules, permit or deny, can be specified in a filter. The relationship between these rules is "OR". This means that if a route meets one of the matching rules, the route matches the AS\_Path filter.

 **NOTE**

For details on a regular expression, see the *Huawei AR150&200 Series Enterprise Routers Configuration Guide - Basic Configurations*.

- Configure a community filter.

A BGP community attribute is used to identify a group of routes with the same properties. Routes can be classified by community attribute. This facilitates route management.

Some AS internal routes may not need to be advertised to any other AS, whereas AS external routes need to be advertised to other ASs. These AS external routes have different prefixes (as a result, an IP prefix list is inapplicable) and may come from different ASs (as a result, an AS\_Path filter is inapplicable). You can set a community attribute value for these AS

internal routes and another community attribute value for these AS external routes on an ASBR to control and filter these routes.

Perform the following steps on a BGP router:

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
ip community-filter
```

A community filter is configured.

- To configure a standard community filter, run the **ip community-filter** { **basic** *comm-filter-name* { **permit** | **deny** } [ *community-number* | *aa:nn* ] \* &<1-9> | *basic-comm-filter-num* { **permit** | **deny** } [ *community-number* | *aa:nn* ] \* &<1-16> } [ **internet** | **no-export-subconfed** | **no-advertise** | **no-export** ] \* command.
- To configure an advanced community filter, run the **ip community-filter** { **advanced** *comm-filter-name* | *adv-comm-filter-num* } { **permit** | **deny** } *regular-expression* command.

- Configure an extcommunity filter.

Similar to a BGP community filter, a BGP extcommunity filter is used to filter private network routes.

Perform the following steps on a BGP router:

1. Run:

```
system-view
```

The system view is displayed.

2. Perform either of the following operations as required to configure an extcommunity filter.

- To configure a basic extcommunity filter, run the **ip extcommunity-filter** { *basic-extcomm-filter-num* | **basic** *basic-extcomm-filter-name* } { **deny** | **permit** } { **rt** { *as-number:nn* | *ipv4-address:nn* } } &<1-16> command.
- To configure an advanced extcommunity filter, run the **ip extcommunity-filter** { *adv-extcomm-filter-num* | **advanced** *adv-extcomm-filter-name* } { **deny** | **permit** } *regular-expression* command.

Multiple entries can be defined in an extcommunity filter. The relationship between the entries is "OR". This means that if a route matches one of the rules, the route matches the filter.

- Configure a route-policy.

A route-policy is used to match routes or route attributes, and to change route attributes when specific conditions are met. As the preceding filters can be used as matching conditions of a route-policy, the route-policy is powerful in functions and can be used flexibly.

Perform the following steps on a BGP router:

1. Run:

**system-view**

The system view is displayed.

2. Run:

```
route-policy route-policy-name { permit | deny } node node
```

A node is configured for a route-policy, and the view of the route-policy is displayed.

A route-policy consists of multiple nodes. For example, the **route-policy route-policy-example permit node 10** command specifies node 10 and the **route-policy route-policy-example deny node 20** command specifies node 20. The two nodes belong to the route-policy specified by **route-policy-example**. The relationship between the nodes of a route-policy is "OR". The details are as follows:

- If a route matches one node, the route matches the route-policy and will not be matched with the next node. For example, there are two nodes defined using the **route-policy route-policy-example permit node 10** and **route-policy route-policy-example deny node 20** commands. If a route matches the node defined using the **route-policy route-policy-example permit node 10** command, the route will not be matched with the node defined using the **route-policy route-policy-example deny node 20** command.
- If a route does not match any node, the route fails to match the route-policy.

When a route-policy is used to filter a route, the route is first matched with the node with the smallest *node* value. For example, if two nodes are configured using the **route-policy route-policy-example permit node 10** and **route-policy route-policy-example deny node 20** commands, a route is first matched with the node configured using the **route-policy route-policy-example permit node 10** command.

 **NOTE**

The AR150/200 considers that each unmatched route fails to match the route-policy by default. If more than one node is defined in a route-policy, at least one of them must be in **permit** mode.

3. (Optional) Perform the following operations as needed to configure **if-match** clauses for current nodes of the route-policy.

**if-match** clauses are used to filter routes. If no **if-match** clause is specified, all routes will match the node in the route-policy.

- To match an ACL, run the **if-match acl** { *acl-number* | *acl-name* } command.
- To match an IP prefix list, run the **if-match ip-prefix** *ip-prefix-name* command.

 **NOTE**

The **if-match acl** and **if-match ip-prefix** commands cannot be used together in the same node of a route-policy, because the latest configuration will override the previous one.

- To match the AS-Path attribute of BGP routes, run the **if-match as-path-filter** { *as-path-filter-number* | *as-path-filter-name* } &<1-16> command.
- To match the community attribute of BGP routes, run either of the following commands:
  - **if-match community-filter** { *basic-comm-filter-num* [ **whole-match** ] | *adv-comm-filter-num* } \* &<1-16>
  - **if-match community-filter** *comm-filter-name* [ **whole-match** ]
- To match the extended community attribute of BGP routes, run the **if-match extcommunity-filter** { { *basic-extcomm-filter-num* | *adv-extcomm-filter-num* } }

<1-16> | *basic-extcomm-filter-name* | *advanced-extcomm-filter-name* }  
 command.

The operations in Step 3 can be performed in any order. A node may have multiple **if-match** clauses or no **if-match** clause.

 **NOTE**

The relationship between the **if-match** clauses in a node of a route-policy is "AND". A route must match all the rules before the action defined by the **apply** clause is taken. For example, if two **if-match** clauses (**if-match acl 2003** and **if-match as-path-filter 100**) are defined in the **route-policy route-policy-example permit node 10** command, a route is considered to match node 10 only when it matches the two **if-match** clauses.

4. (Optional) Perform the following operations as needed to configure **apply** clauses for current nodes of the route-policy:

**apply** clauses can be used to set attributes for routes matching **if-match** clauses. If this step is not performed, the attributes of routes matching **if-match** clauses keep unchanged.

- To replace or add a specified AS number in the AS\_Path attribute of a BGP route, run the **apply as-path as-number** command.
- To delete a specified BGP community attribute from a route, run the **apply comm-filter comm-filter-number delete** command.

 **TIP**

The **apply comm-filter delete** command deletes a specified community attribute from a route. An instance of the **ip community-filter** command can specify only one community attribute each time. To delete more than one community attribute, run the **ip community-filter** command multiple times. If multiple community attributes are specified in one community filter, none of them can be deleted. For more information, see the *Huawei AR150&200 Series Enterprise Routers Command Reference*.

- To delete all community attributes from a BGP route, run the **apply community none** command.
- To set community attributes for a BGP route, run the **apply community { { community-number aa:nn } <1-32> | internet | no-advertise | no-export | no-export-subconfed }\* [ additive ]** command.
- To set an extended community attribute (route-target) for a route, run the **apply extcommunity { rt { as-number:nn | 4as-number:nn | ipv4-address:nn } } <1-16> [ additive ]** command.
- To set the local preference for a BGP route, run the **apply local-preference preference** command.
- To set the Origin attribute for a BGP route, run the **apply origin { igp | egp as-number | incomplete }** command.
- To set a preferred value for a BGP route, run the **apply preferred-value preferred-value** command.
- To set dampening parameters for an EBGP route, run the **apply dampening half-life-reach reuse suppress ceiling** command.

The operations in Step 4 can be performed in any order. A node may have multiple **apply** clauses or no **apply** clause.

----End



## 8.6.3 Configuring to Control the Acceptment of BGP Routing Information

After an import policy is configured, only the routes that match the import policy can be received.

### Procedure

- Configure BGP to receive routes from all its peers or peer groups.

You can configure a BGP device to filter received routes. Perform the following steps on a BGP router:

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
bgp as-number
```

The BGP view is displayed.

3. Run:

```
ipv4-family unicast
```

The IPv4 unicast address family view is displayed.

4. Perform either of the following operations to configure the BGP device to filter the routes received from all its peers or peer groups:

- To filter routes based on a specified ACL, run the **filter-policy { *acl-number* | **acl-name** *acl-name* } import** command.
- To filter routes based on an IP prefix list, run the **filter-policy ip-prefix *ip-prefix-name* import** command.

#### NOTE

If an ACL has been referenced in the **filter-policy** command but no VPN instance is specified in any ACL rule, BGP will filter routes including public network routes and private network routes in all address families. If a VPN instance is specified in an ACL rule, only the data traffic from the VPN instance will be filtered, and no routes of this VPN instance will be filtered.

- Configure a BGP device to receive routes from a specific peer or peer group.

You can configure a BGP device to filter received routes. Perform the following steps on a BGP router:

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
bgp as-number
```

The BGP view is displayed.

3. Run:

```
ipv4-family unicast
```

The IPv4 unicast address family view is displayed.

4. Perform any of the following configurations to configure the BGP device to filter the routes received from a specific peer or peer group:

- To filter routes based on an ACL, run the **peer { ipv4-address | group-name } filter-policy { acl-number | acl-name acl-name } import** command.
- To filter routes based on an IP prefix list, run the **peer { ipv4-address | group-name } ip-prefix ip-prefix-name import** command.
- To filter routes based on an AS\_Path filter, run the **peer { ipv4-address | group-name } as-path-filter { as-path-filter-number | as-path-filter-name } import** command.
- To filter routes based on a route-policy, run the **peer { ipv4-address | group-name } route-policy route-policy-name import** command.

A peer group and its members can use different import policies when receiving routes. This means that each member in a peer group can select its own policy to filter received routes.

- Limit the number of the routes received from a peer or peer group.

When the router running BGP is attacked or network configuration errors occur, the router receives a large number of routes from its neighbor. As a result, a large number of resources of the router are consumed. Therefore, the administrator must limit the resources used by the router based on network planning and the capacity of the router. BGP provides peer-based route control to limit the number of routes to be sent by a neighbor. Thus, the preceding problem is addressed.

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
bgp as-number
```

The BGP view is displayed.

3. Run:

```
ipv4-family unicast
```

The IPv4 unicast address family view is displayed.

4. Run:

```
peer { group-name | ipv4-address } route-limit limit [percentage]
[alert-only | idle-forever | idle-timeout times]
```

The number of routes that can be received from a peer or peer group is set.

The command provides the limit on the number of received routes based on peers. You can configure specific parameters as required to control BGP after the number of the routes received from a peer exceeds the threshold.

- **alert-only**: The peer relationship is kept. No route is received after the number of received routes exceeds the threshold, and an alarm is generated and recorded in the log.
- **idle-forever**: The peer relationship is interrupted. The router does not retry setting up a connection. An alarm is generated and recorded in the log. In this case, run the **display bgp peer [ verbose ]** command, and you can find that the status of the peer is Idle. To restore the BGP connection, run the **reset bgp** command.
- **idle-timeout**: The peer relationship is interrupted. The router retries setting up a connection after the timer expires. An alarm is generated and recorded in the log. In this case, run the **display bgp peer [ verbose ]** command, and you can find that

the status of the peer is Idle. To restore the BGP connection before the timer expires, run the **reset bgp** command.

- If none of the preceding parameters is set, the peer relationship is disconnected. The router retries setting up a connection after 30 seconds. An alarm is generated and recorded in the log.

 **NOTE**

If the number of routes received by the local router exceeds the upper limit and the **peer route-limit** command is used for the first time, the local router and its peer reestablish the peer relationship, regardless of whether **alert-only** is set.

---End

## 8.6.4 Configuring BGP Soft Reset

BGP soft reset allows the system to refresh a BGP routing table dynamically without tearing down any BGP connection if routing policies are changed.

### Context

After changing a BGP import policy, you must reset BGP connections for the new import policy to take effect, interrupting these BGP connections temporarily. BGP route-refresh allows the system to refresh a BGP routing table dynamically without tearing down any BGP connection if routing policies are changed.

- If a device's peer supports route-refresh, the **refresh bgp** command can be used on the device to softly reset the BGP connection with the peer and update the BGP routing table.
- If a device's peer does not support route-refresh, the **peer keep-all-routes** command can be used on the device to remain all routing updates received from the peer so that the device can refresh its routing table without closing the connection with the peer.

Perform the following steps on a BGP router:

### Procedure

- If the device's peers support route-refresh, perform the following operations:
  1. (Optional) Enable route-refresh.
    - a. Run:  
**system-view**  
The system view is displayed.
    - b. Run:  
**bgp as-number**  
The BGP view is displayed.
    - c. Run:  
**peer { ipv4-address | group-name } capability-advertise route-refresh**  
Route-refresh is enabled.

By default, route-refresh is enabled.

If route-refresh is enabled on all BGP routers and the import policy of the local router is changed, the local router sends a route-refresh message to peers or peer groups. After receiving the message, the peers or peer groups resend routing information to the local BGP router. This enables the local router to dynamically

refresh its BGP routing table and apply the new routing policy without closing any BGP connections.

2. Configure BGP soft reset.

- a. Run the **refresh bgp** [ **vpn-instance** *vpn-instance-name* **ipv4-family** ] { **all** | *ipv4-address* | **group** *group-name* | **external** | **internal** } { **export** | **import** } command in the user view to softly reset the BGP connections between the devices and its peers or peer groups.

**external** softly resets an EBGP connection, and **internal** softly resets an IBGP connection.

**export** triggers outbound BGP soft reset, and **import** triggers inbound BGP soft reset.

● If the device's peers do not support route-refresh, perform the following operations:

- Configure the device to store all the routing updates received from its peers or peer groups.

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
bgp as-number
```

The BGP view is displayed.

3. Run:

```
ipv4-family unicast
```

The IPv4 unicast address family view is displayed.

4. Run:

```
peer { ipv4-address | group-name } keep-all-routes
```

The device is configured to store all the routing updates received from its peers or peer groups.

By default, the device stores only the routing updates that are received from peers or peer groups and match a configured import policy.

After this command is used, all routing updates sent by a specified peer or peer group are stored, regardless of whether an import policy is used. When the local routing policy changes, the information can be used to regenerate BGP routes again.

 **NOTE**

This command must be run on the local device and its peers. If the **peer keep-all-routes** command is run on the device for the first time, the sessions between the device and its peers are reestablished.

The **peer keep-all-routes** command does not need to be run on the router that supports route-refresh. If the **peer keep-all-routes** command is run on the router, the sessions between the router and its peers will not be reestablished but the **refresh bgp** command does not take effect on the router.

---End

## 8.6.5 Checking the Configuration

After configuring BGP route reception, you can view the imported routes matching a specified filter.

## Prerequisites

The BGP route reception configurations are complete.

## Procedure

- Run the **display ip as-path-filter** [ *as-path-filter-number* | *as-path-filter-name* ] command to check a configured AS\_Path filter.
- Run the **display ip community-filter** [ *basic-comm-filter-num* | *adv-comm-filter-num* | *comm-filter-name* ] command to check information about a configured community filter.
- Run the **display ip extcommunity-filter** [ *extcomm-filter-number* ] command to check information about a configured extended community filter.
- Run the **display bgp routing-table as-path-filter** { *as-path-filter-number* | *as-path-filter-name* } command to check information about routes matching a specified AS\_Path filter.
- Run the **display bgp routing-table community-filter** { { *community-filter-name* | *basic-community-filter-number* } [ **whole-match** ] | *advanced-community-filter-number* } command to check information about routes matching a specified BGP community filter.
- Run the **display bgp routing-table peer ipv4-address received-routes** [ **active** ] [ **statistics** ] command to check information about routes received by a BGP device from its peers.
- Run the **display bgp routing-table peer ipv4-address accepted-routes** command to check information about the routes that are received by a BGP device from a specified peer and match the routing policy.

---End

## 8.7 Configuring BGP Route Aggregation

Configuring BGP Route Aggregation on a device can reduce the sizes of routing tables on the peers of the device.

### Applicable Environment

The BGP routing table of a device on a medium or large BGP network contains a large number of routing entries. Storing the routing table consumes a large number of memory resources, and transmitting and processing routing information consume lots of network resources. Configuring route aggregation can reduce the size of a routing table, prevent specific routes from being advertised, and minimize the impact of route flapping on network performance. BGP route aggregation and routing policies enable BGP to effectively transmit and control routes.

BGP supports automatic and manual aggregation. Manual aggregation takes precedence over automatic aggregation.

### Pre-configuration Tasks

Before configuring BGP route aggregation, complete the following task:

- [Configuring Basic BGP Functions](#)

### Procedure

- Configure automatic route aggregation.

1. Run:  
**system-view**  
The system view is displayed.

2. Run:  
**bgp as-number**  
The BGP view is displayed.

3. Run:  
**ipv4-family unicast**  
The IPv4 unicast address family view is displayed.

4. Run:  
**summary automatic**  
Automatic aggregation is configured for imported routes.

The **summary automatic** command aggregates routes imported by BGP. The routes can be direct routes, static routes, RIP routes, OSPF routes, or IS-IS routes. After this command is run, BGP aggregates routes based on natural network segments. The command, however, cannot aggregate routes imported using the **network** command.

● Configure manual route aggregation.

1. Run:  
**system-view**  
The system view is displayed.

2. Run:  
**bgp as-number**  
The BGP view is displayed.

3. Run:  
**ipv4-familyunicast**  
The IPv4 unicast address family view is displayed.

4. Run:  
**aggregate ipv4-address { mask | mask-length } [ as-set | attribute-policy route-policy-name1 | detail-suppressed | origin-policy route-policy-name2 | suppress-policy route-policy-name3 ] \***

Manual route aggregation is configured.

**as-set** is used to generate an aggregated route in which the AS\_Path attribute contains AS\_Path information of specific routes. If many routes need to be aggregated, exercise caution when using this parameter. Frequent changes in specific routes cause flapping of the aggregated route.

**detail-suppressed** is used to suppress the advertisement of specific routes. After **detail-suppressed** is set, only aggregated routes are advertised. Aggregated routes carry the atomic-aggregate attribute, not the community attributes of specific routes.

**suppress-policy** is used to suppress the advertisement of specified routes. The **if-match** clause of **route-policy** can be used to filter routes to be suppressed. Only the routes matching the policy will be suppressed, and the other routes will still be advertised. The **peer route-policy** command can also be used to filter out the routes not to be advertised to peers.

After **origin-policy** is used, only the routes matching **route-policy** are aggregated.

**attribute-policy** is used to set attributes for an aggregated route. If the **AS\_Path** attribute is set in the policy using the **apply as-path** command and **as-set** is set in the **aggregate** command, the **AS\_Path** attribute in the policy does not take effect. The **peer route-policy** command can also be used to set attributes for an aggregated route.

Only the routes that exist in the local BGP routing table can be manually aggregated. For example, if route 10.1.1.1/24 is not in the BGP routing table, BGP will not generate an aggregated route for it even if the **aggregate 10.1.1.1 16** command is used.

When using manual aggregation, you can apply various routing policies and set route attributes.

---End

## Checking the Configuration

After route aggregation is configured, you can check whether the configuration is correct.

- Run the **display bgp routing-table [ network [ mask | mask-length ] ]** command to check information about BGP aggregated routes.

## 8.8 Configuring BGP Peer Groups

Configuring BGP peer groups simplifies the BGP network configuration and improves the route advertisement efficiency.

### 8.8.1 Establishing the Configuration Task

Before configuring BGP peer groups, familiarize yourself with the applicable environment, complete the pre-configuration tasks, and obtain the data required for the configuration. This will help you complete the configuration task quickly and efficiently.

#### Applicable Environment

A BGP peer group consists of BGP peers that have the same update policies and configurations.

A large-scale BGP network has a large number of peers. Configuring and maintaining these peers is difficult. To address this problem, configure a BGP peer group for BGP peers with the same configurations. Configuring BGP peer groups simplifies peer management and improves the route advertisement efficiency.

Based on the ASs where peers reside, peer groups are classified as follows:

- IBGP peer group: The peers of an IBGP peer group are in the same AS.
- Pure EBGP peer group: The peers of a pure EBGP peer group are in the same external AS.
- Mixed EBGP peer group: The peers of a mixed EBGP peer group are in different external ASs.

If a function is configured on a peer and its peer group, the function configured on the peer takes precedence over that configured on the peer group. After a peer group is created, peers can be added to the peer group. If these peers are not configured separately, they will inherit the configurations of the peer group. If a peer in a peer group has a specific configuration

requirement, the peer can be configured separately. The configuration of this peer will override the configuration inherited by the peer from the peer group.

## Pre-configuration Tasks

Before configuring BGP peer groups, complete the following task:

- **Configuring Basic BGP Functions**

## Data Preparation

To configure BGP peer groups, you need the following data.

| No. | Data                                                                  |
|-----|-----------------------------------------------------------------------|
| 1   | Type and name of a peer group, and IP addresses of peer group members |

## 8.8.2 Creating IBGP Peer Groups

If multiple IBGP peers exist, adding them to an IBGP peer group can simplify the BGP network configuration and management. When creating an IBGP peer group, you do not need to specify an AS number for the IBGP peer group.

### Procedure

**Step 1** Run:

```
system-view
```

The system view is displayed.

**Step 2** Run:

```
bgp as-number
```

The BGP view is displayed.

**Step 3** Run:

```
group group-name internal
```

An IBGP peer group is created.

**Step 4** Run:

```
peer ipv4-address group group-name
```

A peer is added to the peer group.

#### NOTE

You can repeat step 4 to add multiple peers to the peer group. If the local device has not established a peer relationship with this peer, the device will attempt to establish a peer relationship with this peer, and set the AS number of this peer to the AS number of the peer group.

When creating an IBGP peer group, you do not need to specify the AS number.



After configuring a peer group, you can configure BGP functions for the peer group. By default, all peers in a peer group inherit the entire configuration of the peer group. The inherited configuration can be overridden if you directly configure commands for the peer.

----End

### 8.8.3 Creating Pure EBGP Peer Groups

If multiple EBGP peers exist in an AS, adding them to an EBGP peer group can simplify the BGP network configuration and management. All the peers in a pure EBGP peer group must have the same AS number.

#### Procedure

**Step 1** Run:

```
system-view
```

The system view is displayed.

**Step 2** Run:

```
bgp as-number
```

The BGP view is displayed.

**Step 3** Run:

```
group group-name external
```

A pure EBGP peer group is created.

**Step 4** Run:

```
peer group-name as-number as-number
```

An AS number is set for the EBGP peer group. If peers already exist in a peer group, you can neither change the AS number of the peer group nor delete the AS number of the peer group by using the **undo peer as-number** command.

**Step 5** Run:

```
peer ipv4-address group group-name
```

A peer is added to the peer group.

#### NOTE

You can repeat step 5 to add multiple peers to the peer group. If the local device has not established a peer relationship with this peer, the device will attempt to establish a peer relationship with this peer, and set the AS number of this peer to the AS number of the peer group.

After configuring a peer group, you can configure BGP functions for the peer group. By default, all peers in a peer group inherit the entire configuration of the peer group. The inherited configuration can be overridden if you directly configure commands for the peer.

----End

### 8.8.4 Creating Mixed EBGP Peer Groups

If multiple EBGP peers exist in different ASs, adding them to a mixed EBGP peer group can simplify the BGP network configuration and management. When creating a mixed EBGP peer group, you need to specify an AS number for each peer.

## Procedure

**Step 1** Run:

```
system-view
```

The system view is displayed.

**Step 2** Run:

```
bgp as-number
```

The BGP view is displayed.

**Step 3** Run:

```
group group-name external
```

A mixed EBGP peer group is created.

**Step 4** Run:

```
peer ipv4-address as-number as-number
```

A peer is created and an AS number is set for this peer.

**Step 5** Run:

```
peer ipv4-address group group-name
```

The peer is added to the peer group.

 **NOTE**

You can repeat Steps 4 and 5 to add multiple peers to the peer group.

You need to specify an AS number for each peer in a mixed EBGP peer group.

After configuring a peer group, you can configure BGP functions for the peer group. By default, all peers in a peer group inherit the entire configuration of the peer group. The inherited configuration can be overridden if you directly configure commands for the peer.

----End

## 8.8.5 Checking the Configuration

After BGP peer groups are configured, you can view information about BGP peers and BGP peer groups.

### Prerequisites

The BGP peer group configurations are complete.

### Procedure

- Run the **display bgp peer** [ *ipv4-address* ] **verbose** command to check detailed information about BGP peers.
- Run the **display bgp group** [ *group-name* ] command to check information about BGP peer groups.

 **NOTE**

This command is applied only to devices on which BGP peer groups are created.

If a peer group is specified in this command, detailed information about this peer group will be displayed. If no peer group is specified in this command, information about all BGP peer groups is displayed.

---End

## 8.9 Configuring BGP Route Reflectors

Deploying BGP RRs allows IBGP peers to communicate without establishing full-mesh connections between them. Using BGP RRs simplifies network configurations and improves route advertisement efficiency.

### 8.9.1 Establishing the Configuration Task

Before configuring BGP RRs, familiarize yourself with the applicable environment, complete the pre-configuration tasks, and obtain the data required for the configuration. This will help you complete the configuration task quickly and efficiently.

#### Applicable Environment

BGP uses the AS\_Path attribute to prevent route loops, but it does not change the AS\_Path attribute of a route sent between IBGP peers within an AS. This may cause a route loop. To prevent this problem, the BGP standard defines that a BGP device is prohibited from advertising any route that received from another IBGP peer. Full-mesh connections then must be created between IBGP peers to ensure the connectivity between them. If many IBGP peers exist, the overhead will be large and the configuration workload will be heavy for establishing full-mesh logical connections between routers. In addition, the network will be difficult to maintain.

Using BGP confederations or RRs can solve these problems. A BGP confederation consists of several sub-ASs in an AS. Full-mesh logical connections need to be established and maintained between IBGP peers in each sub-AS. To deploy RRs, you only need to configure the RR functionality on routers and do not need to change configurations on other devices. In this regard, deploying RRs is easier and more flexible than deploying confederations.

#### Pre-configuration Tasks

Before configuring a BGP RR, complete the following task:

- **Configuring Basic BGP Functions**

#### Data Preparation

To configure a BGP RR, you need the following data.

| No. | Data                                            |
|-----|-------------------------------------------------|
| 1   | Role of each router (RR, client, or non-client) |
| 2   | (Optional) Cluster ID of the RR                 |

## 8.9.2 Configuring a Route Reflector and Specifying Clients

Deploying an RR and clients in an address family allows IBGP peers to communicate without having full-mesh logical connections established between them, reducing network configuration and maintenance workload, and improving network performance.

### Context

In an AS, one router serves as an RR, and the other routers serve as clients. IBGP peer relationships are set up between the RR and clients. The RR reflects routes between clients, and BGP connections do not need to be established between the clients. A BGP device that is neither an RR nor a client is called a non-client. Non-clients and the RR must establish full-mesh connections with each other.

After receiving IBGP routes, the RR selects optimal routes based on BGP route selection policies and advertises learned routes to its clients and non-clients following the rules described below:

- After learning routes from non-clients, the RR advertises the routes to all clients.
- After learning routes from clients, the RR advertises the routes to all non-clients and clients.

In addition, the RR advertises learned EBGP routes to all non-clients and clients.

It is easy to configure an RR. The RR functionality only needs to be configured on one router. Configurations on clients are not required.

Perform the following steps on the router that is running BGP and is to be specified as an RR:

### Procedure

**Step 1** Run:

```
system-view
```

The system view is displayed.

**Step 2** Run:

```
bgp as-number
```

The BGP view is displayed.

**Step 3** Run:

```
ipv4-family unicast
```

The IPv4 unicast address family view is displayed.

**Step 4** Run:

```
peer { ipv4-address | group-name } reflect-client
```

The router is specified as an RR and its clients are configured.

To add more clients, repeat the step.

**reflect-client** configured in an address family is valid only in this address family and cannot be inherited by other address families. Configuring **reflect-client** in a specified address family is recommended.

---End

## 8.9.3 (Optional) Disabling Route Reflection Between Clients

If the clients of an RR are fully meshed, prohibiting route reflection among the clients can reduce the link cost.

### Context

The RR usually advertises the routes learned from clients to all non-clients and clients. If full-mesh logical connections have been established between all the clients of the RR, the clients are capable of sending routes to each other without the help of the RR. Route reflection can be disabled between clients to reduce the stress on the RR.

Perform the following steps on the RR that is running BGP.

### Procedure

**Step 1** Run:

```
system-view
```

The system view is displayed.

**Step 2** Run:

```
bgp as-number
```

The BGP view is displayed.

**Step 3** Run:

```
ipv4-family unicast
```

The IPv4 unicast address family view is displayed.

**Step 4** Run:

```
undo reflect between-clients
```

Route reflection is disabled between clients.

If the clients of an RR have established full-mesh connections with each other, the **undo reflect between-clients** command can be used to disable route reflection between clients in order to reduce the link cost. By default, route reflection is enabled between the clients of an RR.

This command can only be configured on the RR.

----End

## 8.9.4 (Optional) Configuring the Cluster ID for a Route Reflector

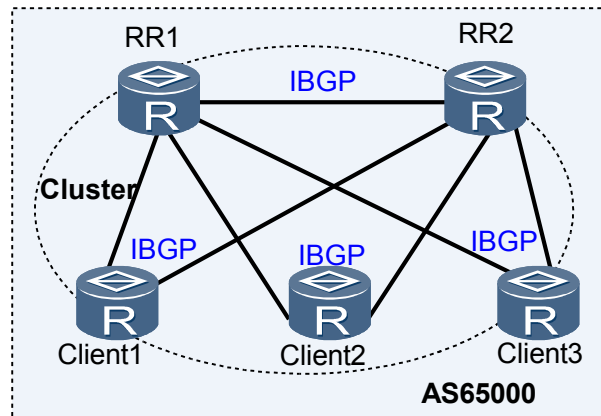
If several RRs are deployed in a cluster, assigning the same cluster ID to them can prevent route loops.

### Context

A backup RR is usually deployed in an AS to prevent a fault on an RR from causing the clients and non-clients unable to receive routing information. This backup RR improves network reliability.

As shown in [Figure 8-1](#), RR1 and RR2 are configured as backups for each other in AS 65000. Clients 1, 2, and 3 are their clients. An IBGP peer relationship is set up between RR1 and RR2 so that each RR is the other RR's non-client.

Figure 8-1 RR cluster



Route loops may easily occur in this network. For example, when Client1 receives an updated route from an EBGP peer, it uses IBGP to advertise this route to RR1 and RR2. Then the following problems will happen in the same time:

- RR1 advertises it to its clients and non-client (RR2),
- RR2 advertises it to its clients and non-client (RR1).

As a result, a route loop occurs between RR1 and RR2.

To address this problem, configure all routers on the network shown in [Figure 8-1](#) into the same cluster and assign them the same cluster ID. After the configuration is complete, if Client1 receives an updated route from an EBGP peer, it uses IBGP to advertise this route to RR1 and RR2.

- After receiving this route, RR1 reflects it to its clients and RR2 and adds the local cluster ID to the front of the cluster list.
- After receiving the route reflected from RR1, RR2 checks the cluster list. After finding that the local cluster ID is already on the cluster list, RR2 discards the route.

**NOTE**

Using a cluster list prevents route loops between RRs within an AS.

Perform the following steps on each router that is running BGP:

## Procedure

**Step 1** Run:

```
system-view
```

The system view is displayed.

**Step 2** Run:

```
bgp as-number
```

The BGP view is displayed.

**Step 3** Run:

```
ipv4-family unicast
```

The IPv4 unicast address family view is displayed.

**Step 4** Run:

```
reflector cluster-id cluster-id
```

A cluster ID is configured.

If a cluster has multiple RRs, use this command to set the same *cluster-id* for these RRs to prevent route loops.

 **NOTE**

To ensure that a client can learn the routes reflected by an RR, the Cluster ID configured on the RR must be different from the Cluster ID of the client (By default, the client uses its Router ID as the cluster ID). If the Cluster ID is the same as the Cluster ID of the client, the client discards received routes.

----End

## 8.9.5 (Optional) Preventing BGP Routes from Being Added into the IP Routing Table

Disabling BGP route delivery to the IP routing table on an RR can prevent traffic from being forwarded by the RR, improving route advertisement efficiency.

### Context

Usually, BGP routes are delivered to the IP routing table on the router to guide traffic forwarding. If the router does not need to forward traffic, disable BGP route delivery to the IP routing table on the router.

BGP route delivery to the IP routing table is generally disabled on RRs. An RR transmits routes and forwards traffic within an AS. If the RR is connected to many clients and non-clients, the route transmission task will consume a lot of CPU resources of the RR and cause the RR unable to implement traffic forwarding. To improve the efficiency of route transmission, disable BGP route delivery to the IP routing table on the RR to make the RR dedicated to route transmission.

Perform the following steps on the router that is running BGP.

### Procedure

**Step 1** Run:

```
system-view
```

The system view is displayed.

**Step 2** Run:

```
bgp as-number
```

The BGP view is displayed.

**Step 3** Run:

```
ipv4-family unicast
```

The IPv4 unicast address family view is displayed.

**Step 4** Run:

```
bgp-rib-only [route-policy route-policy-name]
```

BGP route delivery to the IP routing table is disabled.

The routes preferred by BGP are delivered to the IP routing table by default.

If **route-policy** *route-policy-name* is configured in the **bgp-rib-only** command, routes matching the policy are not delivered to the IP routing table, and routes not matching the policy are delivered to the IP routing table, with the route attributes unchanged.

 **NOTE**

The **bgp-rib-only** command and the **active-route-advertise** command are mutually exclusive.

----End

## 8.9.6 Checking the Configuration

After configuring BGP RRs, you can view BGP RR configurations and routing information transmitted by BGP.

### Prerequisites

All BGP RR configurations are complete.

### Procedure

- Run the **display bgp [ vpnv4 [ vpn-instance vpn-instance-name | all ] ] peer [ ipv4-address ] verbose** command to check detailed information about BGP peers.
- Run the **display bgp routing-table [ network [ { mask | mask-length } [ longer-prefixes ] ] ]** command to check information in a BGP routing table.

----End

## 8.10 Configuring a BGP Confederation

BGP confederations can be configured on a large BGP network to reduce the number of IBGP connections and simplify routing policy management, increasing route advertisement efficiency.

### Applicable Environment

A confederations can be used to reduce the number of IBGP connections in an AS. It divides an AS into several sub-ASs. Full-mesh IBGP connections are established between devices in each sub-AS, and full-mesh EBGP connections are established between devices in different sub-ASs,

Compared with RRs, confederations facilitate IGP extensions.

### Pre-configuration Tasks

Before configuring a BGP confederation, complete the following tasks:

- Configuring link layer protocol parameters for interfaces to ensure that the link layer protocol on the interfaces is Up
- [Configuring Basic BGP Functions](#)

### Procedure

- Configure a BGP confederation.

Perform the following steps on a BGP device:



1. Run:  
`system-view`  
The system view is displayed.
2. Run:  
`bgp as-number`  
The BGP view is displayed.
3. Run:  
`confederation id as-number`  
A confederation ID is set.
4. Run:  
`confederation peer-as as-number &<1-32>`  
The number of the sub-AS where other EBGp peers connected to the local AS reside is set.  
  
*as-number* is valid in the confederation only when the sub-ASs of the confederation are configured.  
  
The **confederation id** and **confederation peer-as** commands must be run on all the EBGp peers in the same confederation, and the same confederation ID must be set for these EBGp peers.

 **NOTE**

An old speaker that has a 2-byte AS number cannot be in the same confederation with a new speaker that has a 4-byte AS number. Otherwise, a routing loop may occur. This is because the AS4\_Path attribute does not support confederations.

- Configure confederation compatibility.

Other routers may implement the confederation that does not comply with the RFC standard. In such a situation, confederation compatibility must be configured. Perform the following steps on a BGP device:

1. Run:  
`system-view`  
The system view is displayed.
2. Run:  
`bgp as-number`  
The BGP view is displayed.
3. Run:  
`confederation nonstandard`  
The routers are configured to be compatible with the nonstandard AS confederation.  
By default, the configured confederation accords with RFC 3065.

---End

## Checking the Configuration

After a confederation is configured, you can check whether the configuration is correct.

- Run the **display bgp peer** [ *ipv4-address* ] **verbose** command to check detailed information about BGP peers.
- Run the **display bgp routing-table** [ *network* [ { *mask* | *mask-length* } ] [ **longer-prefixes** ] ] ] command to check routing information in a BGP routing table.

## 8.11 Configuring BGP Community Attributes

Community attributes are used to simplify routing policy management.

### 8.11.1 Establishing the Configuration Task

Before configuring BGP community attributes, familiarize yourself with the applicable environment, complete the pre-configuration tasks, and obtain the data required for the configuration. This will help you complete the configuration task quickly and accurately.

#### Applicable Environment

Community attributes are used to simplify routing policy application and facilitate network maintenance. They allow a group of BGP routers in different ASs to share the same routing policies. Before advertising a route with the community attribute to peers, a BGP router can change the original community attribute of this route. Community attributes are route attributes, which are transmitted between BGP peers, and the transmission is not restricted within an AS.

#### Pre-configuration Tasks

Before configuring BGP community attributes, complete the following task:

- [Configuring Basic BGP Functions](#)

#### Data Preparation

To configure BGP Community attributes, you need the following data.

| No. | Data                                                            |
|-----|-----------------------------------------------------------------|
| 1   | Community attribute value                                       |
| 2   | Route-policy name, node sequence number, and matching condition |
| 3   | Names of inbound and outbound routing policies                  |

### 8.11.2 Configuring Community Attribute-Related Routing Policies

A routing policy that references a community attribute needs to be configured before the community attribute is advertised.

#### Procedure

- Step 1** Run:
- ```
system-view
```

The system view is displayed.

Step 2 Run:

```
route-policy route-policy-name { permit | deny } node node
```

A node is configured for a routing policy, and the view of the routing policy is displayed.

Step 3 (Optional) Configure filtering conditions (if-match clauses) for a routing policy. Community attributes can be added only to the routes that pass the filtering, and the community attributes of only the routes that pass the filtering can be modified.

For configuration details, see [\(Optional\) Configuring if-match Clauses](#).

Step 4 Configure community or extended community attributes for BGP routes.

● Run:

```
apply community { { community-number | aa:nn } &<1-32> | internet | no-  
advertise | no-export | no-export-subconfed }* [ additive ]
```

Community attributes are configured for BGP routes.

 **NOTE**

A maximum of 32 community attributes can be configured in the **apply community** command.

● Run:

```
apply extcommunity { rt { as-number:nn | ipv4-address:nn } } &<1-16>  
[ additive ]
```

An extended community attribute (Route-Target) is configured for BGP routes.

----End

8.11.3 Configuring a BGP Device to Send Community Attributes to Its Peer

A community attribute takes effect only after the community attribute and the routing policy referencing the community attribute are advertised.

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
bgp as-number
```

The BGP view is displayed.

Step 3 Run:

```
ipv4-family unicast
```

The IPv4 unicast address family view is displayed.

Step 4 Run:

```
peer { ipv4-address | group-name } route-policy route-policy-name export
```

An export routing policy is configured.

 **NOTE**

When configuring a BGP community, use a routing policy to define the community attribute, and apply the routing policy to the routes to be advertised.

For details on routing policy configurations, see the chapter "Routing Policy Configuration."

Step 5 Run one of the following commands as needed to configure a BGP device to advertise community attributes to its peer or peer group.

- To configure the BGP device to send a standard community attribute to its peer or peer group, run:

```
peer { ipv4-address | group-name } advertise-community
```

By default, a device advertises no community attribute to its peer or peer group.

- To configure the BGP device to send an extended community attribute to its peer or peer group, run:

```
peer { ipv4-address | group-name } advertise-ext-community
```

By default, a device advertises no extended community attribute to its peer or peer group.

----End

8.11.4 Checking the Configuration

After configuring BGP community attributes, you can view the configured BGP community attributes.

Prerequisites

The BGP community attribute configurations are complete.

Procedure

- Run the **display bgp routing-table** *network* [*mask* | *mask-length*] command to check the detailed information about BGP routes.
- Run the **display bgp routing-table community** [*community-number* | *aa:nn*] &<1-29> [**internet** | **no-advertise** | **no-export** | **no-export-subconfed**] * [**whole-match**] command to check information about the routes carrying specified BGP community attributes.

----End

8.12 Configuring Prefix-based BGP ORF

Prefix-based BGP ORF is used to enable a BGP device to send to its BGP peer a set of routing policies that can be used by its peer to filter out unwanted routes during route advertisement.

Applicable Environment

During routing information transmission between two devices, routing policies can be used on receiving and sending devices to filter routes.

- If a routing policy is used to filter routing information received by the route receiving device but no policy is used to filter routing information to be sent by the route sending device and the route sending device sends a great deal of routing information, the route receiving device will have to process a great deal of unwanted routing information. This consumes a lot of network bandwidth resources.

- If routes to be advertised by the route sending device need to be filtered and the device has many BGP peers, many export policies need to be configured on the device. This is unhelpful for network planning and maintenance and consumes lots of memory resources.

To address these problems, prefix-based BGP ORF is used to implement on-demand BGP route advertisement. A BGP device uses an export policy provided by a route receiving device to filter routes before sending these routes. It is unnecessary for the local device to provide a separate export policy for each BGP peer. As a result, the loads of the two communication devices, network bandwidth consumption, and configuration workload are reduced.

 **NOTE**

Currently, only prefix-based export policies are supported.

Pre-configuration Tasks

Before configuring prefix-based BGP ORF, complete the following tasks:

- [Configuring Basic BGP Functions](#)
- [Configuring an IPv4 Prefix List](#)

Data Preparation

To configure prefix-based BGP ORF, you need the following data.

No.	Data
1	Address of a peer or name of a peer group
2	Name of an IP prefix list

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
bgp as-number
```

The BGP view is displayed.

Step 3 Run:

```
ipv4-family unicast
```

The IPv4 unicast address family view is displayed.

Step 4 Run:

```
peer { group-name | ipv4-address } capability-advertise orf [ cisco-compatible ] ip-prefix { both | receive | send }
```

Prefix-based ORF is enabled for a BGP peer or peer group.

By default, prefix-based ORF is not enabled for a peer or peer group.

 **NOTE**

This step needs to be performed on both communication devices.

The ORF capability supported by non-Huawei devices may be different from that defined in the RFC standard. To enable a Huawei device to communicate with a non-Huawei device, ensure that the devices are configured with the same compatibility mode (either Cisco-compatible or RFC-compatible). By default, the RFC-compatible mode is used.

BGP ORF has three modes: send, receive, and both. In send mode, a device can send ORF information. In receive mode, a device can receive ORF information. In both mode, a device can either send or receive ORF information. To enable a device to receive ORF IP-prefix information, configure the both or receive mode on the device and the both or send mode on its peer.

Step 5 Run:

```
peer { group-name | ipv4-address } ip-prefix ip-prefix-name import
```

A prefix-based import policy is configured for a peer or peer group.

 **NOTE**

This step is performed only on the receiving device. An IP prefix list specified by *ip-prefix-name* must have been configured. Otherwise, route filtering cannot be implemented. For details on IPv4 prefix list configurations, see [10.3.2 Configuring an IPv4 Prefix List](#).

----End

Checking the Configuration

After prefix-based BGP ORF is configured, you can run the following commands to check the previous configuration.

- Run the **display bgp peer [ipv4-address] verbose** command to check prefix-based BGP ORF negotiation information.
- Run the **display bgp peer ipv4-address orf ip-prefix** command to check prefix-based BGP ORF information received from a specified peer.

 **NOTE**

The **display bgp peer ipv4-address orf ip-prefix** command must be run only on devices that have sent routing information.

8.13 Configuring to Adjust the BGP Network Convergence Speed

You can adjust the BGP network convergence speed by adjusting BGP peer connection parameters to adapt to changes on large-scale networks.

8.13.1 Establishing the Configuration Task

Before adjusting the BGP network convergence speed, familiarize yourself with the applicable environment, complete the pre-configuration tasks, and obtain the data required for the configuration. This will help you complete the configuration task quickly and efficiently.

Applicable Environment

BGP is used to transmit routing information on large-scale networks. Frequent network changes affect the establishment and maintenance of BGP peer relationships, affecting the BGP network convergence speed.

The route dampening and triggered update functions of BGP suppress frequent route changes to a certain extent, but cannot minimize the impact of network flapping on BGP connections.

You can configure BGP timers, disabling rapid EBGP connection reset, and enable BGP tracking to suppress BGP network flapping and speed up BGP network convergence.

- ConnectRetry timer

A ConnectRetry timer is used to set an interval between BGP attempts to initiate TCP connections. After BGP initiates a TCP connection, the ConnectRetry timer will be stopped if the TCP connection is established successfully. If the first attempt to establish a TCP connection fails, BGP tries again to establish the TCP connection after the ConnectRetry timer expires.

You can accelerate or slow down the establishment of BGP peer relationships by changing the BGP ConnectRetry interval. For example, if the ConnectRetry interval is reduced, BGP will wait less time before retrying to establish a TCP connection when the previous attempt fails. This speeds up TCP connection establishment. If a BGP peer flaps constantly, the ConnectRetry interval can be increased to suppress route flapping caused by BGP peer flapping. This speeds up route convergence.

- BGP Keepalive and hold timers

BGP uses Keepalive messages to maintain BGP peer relationships and monitor connection status.

After establishing a BGP connection, two peers send Keepalive messages periodically to each other to detect the BGP connection status. If the router does not receive any Keepalive message or any other types of packets from the peer within the hold time, the router considers the BGP connection interrupted and closes the BGP connection.

- BGP MinRouteAdvertisementIntervalTimer

BGP does not periodically update a routing table. When BGP routes change, BGP updates the changed BGP routes in the BGP routing table by sending Update messages. If a route changes frequently, to prevent the router from sending Update messages upon every change, set the interval at which Update messages are sent.

- Rapid EBGP connection reset

Rapid EBGP connection reset is enabled by default so that EBGP can quickly detect the status of interfaces used to establish EBGP connections. If the interface status is changed frequently, rapid EBGP connection reset can be disabled. As a result, direct EBGP sessions will not be reestablished and deleted as interface alternates between Up and Down. This implements rapid network convergence.

- BGP tracking

BGP tracking can speed up network convergence by adjusting the interval between peer unreachability discovery and connection interruption. BGP tracking is easy to deploy and has good extensibility.

Pre-configuration Tasks

Before adjusting the BGP network convergence speed, complete the following tasks:

- **Configuring Basic BGP Functions**

Data Preparation

To adjust the BGP network convergence speed, you need the following data.

No.	Data
1	Value of the ConnectRetry timer
2	Values of BGP Keepalive and hold timers
3	Value of the MinRouteAdvertisementIntervalTimer
4	Interval between peer unreachability discovery and connection interruption

8.13.2 Configuring a BGP ConnectRetry Timer

You can control the speed at which BGP peer relationships are established by changing the BGP ConnectRetry timer value.

Context

After BGP initiates a TCP connection, the ConnectRetry timer will be stopped if the TCP connection is established successfully. If the first attempt to establish a TCP connection fails, BGP tries again to establish the TCP connection after the ConnectRetry timer expires.

- Setting a short ConnectRetry interval reduces the period BGP waits between attempts to establish a TCP connection. This speeds up the establishment of the TCP connection.
- Setting a long connectRetry interval suppresses routing flapping caused by peer relationship flapping.

A ConnectRetry timer can be configured either for all peers or peer groups, or for a specific peer or peer group. A ConnectRetry timer configured for a specific peer takes precedence over that configured for the peer group of this peer. In addition, a ConnectRetry timer configured for a specific peer or peer group takes precedence over that configured for all peers or peer groups.

Procedure

- Configure a BGP ConnectRetry timer for all peers or peer groups.

Perform the following steps on a BGP router:

1. Run:
`system-view`
The system view is displayed.
2. Run:
`bgp as-number`
The BGP view is displayed.
3. Run:
`timer connect-retry connect-retry-time`

A BGP ConnectRetry timer is configured for all peers or peer groups.

By default, the ConnectRetry timer value is 32s.

- Configure a ConnectRetry timer for a specific peer or peer group.

Perform the following steps on a BGP router:

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
bgp as-number
```

The BGP view is displayed.

3. Run:

```
peer { group-name | ipv4-address } timer connect-retry connect-retry-time
```

A ConnectRetry timer is configured for a specific peer or peer group.

By default, the ConnectRetry timer value is 32s.

The ConnectRetry timer configured for a peer or peer group takes precedence over that configured for all peers or peer groups.

----End

8.13.3 Configuring BGP Keepalive and Hold Timers

The values of BGP Keepalive and hold timers determine the speed at which BGP detects network faults. You can adjust the values of these timers to improve network performance.

Context

Keepalive messages are used by BGP to maintain peer relationships. After establishing a BGP connection, two peers periodically send Keepalive messages to each other to detect BGP peer relationship status. If a device receives no Keepalive message from its peer after the hold timer expires, the device considers the BGP connection to be closed.

- If short Keepalive time and holdtime are set, BGP can detect a link fault quickly. This speeds up BGP network convergence, but increases the number of Keepalive messages on the network and loads of routers, and consumes more network bandwidth resources.
- If long Keepalive time and holdtime are set, the number of Keepalive messages on the network is reduced. This reduces loads of routers. If the Keepalive time is too long, BGP is unable to detect link status changes in a timely manner. This is unhelpful for implementing rapid BGP network convergence and may cause many packets to be lost.



CAUTION

Changing timer values using the **timer** command or the **peer timer** command interrupts BGP peer relationships between routers. Therefore, exercise caution before changing the value of a timer.

Keepalive and hold timers can be configured either for all peers or peer groups, or for a specific peer or peer group. Keepalive and hold timers configured for a specific peer take precedence

over those configured for the peer group of this peer. In addition, Keepalive and hold timers configured for a specific peer or peer group take precedence over those configured for all peers or peer groups.

Procedure

- Configure BGP timers for all peers or peer groups.

Perform the following steps on a BGP router:

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
bgp as-number
```

The BGP view is displayed.

3. Run:

```
timer keepalive keepalive-time hold hold-time
```

BGP timers are configured.

The proper maximum interval at which Keepalive messages are sent is one third the holdtime and is not less than one second. If the holdtime is not set to 0, it is 3s at least. By default, the *keepalive-time* value is 60s and the *hold-time* value is 180s.

NOTE

Setting the Keepalive time to 20s is recommended. If the Keepalive time is smaller than 20s, sessions between peers may be closed.

When setting values of *keepalive-time* and *hold-time*, note the following points:

- The *keepalive-time* and *hold-time* values cannot be both set to 0. Otherwise, the BGP timers become invalid, meaning that BGP will not send Keepalive messages to detect connection status.
- The *hold-time* value cannot be much greater than the *keepalive-time* value. For example, *keepalive-time* cannot be set to 1 while *hold-time* is set to 65535. If the *hold-time* value is too large, BGP cannot detect connection status in time.

After a connection is established between peers, the *keepalive-time* and *hold-time* values are negotiated by the peers. The smaller one of the *hold-time* values carried by Open messages of both peers is taken as the *hold-time* value. The smaller of one third of the *hold-time* value and the locally configured *keepalive-time* value is taken as the *keepalive-time* value.

- Configure timers for a specific peer or peer group.

Perform the following steps on a BGP router:

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
bgp as-number
```

The BGP view is displayed.

3. Run:

```
peer { ipv4-address | group-name } timer keepalive keepalive-time hold  
hold-time
```

The Keepalive and hold timer values are set for a specific peer or peer group.

For information about the relationship between the *keepalive-time* and *hold-time* values, see [Configure BGP timers for all peers or peer groups](#).

 **NOTE**

Setting the Keepalive time to 20s is recommended. If the Keepalive time is smaller than 20s, sessions between peers may be closed.

Timers set for a specific peer or peer group takes precedence over timers set for all peers or peer groups.

---End

8.13.4 Configuring a MinRouteAdvertisementIntervalTimer

A proper MinRouteAdvertisementIntervalTimer can be configured to suppress frequent route changes, improving BGP network stability.

Context

BGP peers use update messages to exchange routing information. Update messages can be used to advertise multiple reachable routes with the same attributes or withdraw multiple unreachable routes.

BGP does not periodically update a routing table. When BGP routes change, BGP updates the changed BGP routes in the BGP routing table by sending Update messages. If a route changes frequently, to prevent the router from sending Update messages upon every change, set the interval at which Update messages are sent.

Perform the following steps on a BGP router:

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
bgp as-number
```

The BGP view is displayed.

Step 3 Run:

```
peer { ipv4-address | group-name } route-update-interval interval
```

A MinRouteAdvertisementIntervalTimer is configured.

By default, the interval at which Update messages are sent to IBGP peers is 15s, and the interval at which Update messages are sent to EBGP peers is 30s.

ipv4-address specifies the address of a specific group. *group-name* specifies the name of a peer group. The `MinRouteAdvertisementIntervalTimer` configured for a peer takes precedence over the `MinRouteAdvertisementIntervalTimer` configured for a peer group.

----End

8.13.5 Disabling Fast Reset of EBGP Connections

Disabling rapid EBGP connection reset can prevent repeated reestablishment and deletion of EBGP sessions in the event of route flapping. This speeds up BGP network convergence.

Context

Rapid EBGP connection reset is enabled by default. This allows BGP to immediately respond to a fault on an interface and delete the direct EBGP sessions on the interface without waiting for the hold timer to expire and implements rapid BGP network convergence.

NOTE

Rapid EBGP connection reset enables BGP to quickly respond to interface faults but does not enable BGP to quickly respond to interface recovery. After the interface recovers, BGP uses its state machine to restore relevant sessions.

If the status of an interface used to establish an EBGP connection changes frequently, the EBGP session will be deleted and reestablished repeatedly, causing network flapping. Rapid EBGP connection reset can be disabled in such a situation. BGP will delete direct EBGP sessions on the interface until the hold timer expires. This suppresses BGP network flapping, helps to implement rapid BGP network convergence, and reduces network bandwidth consumption.

Perform the following steps on a BGP router.

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
bgp as-number
```

The BGP view is displayed.

Step 3 Run:

```
undo ebgp-interface-sensitive
```

Rapid EBGP connection reset is disabled.

NOTE

Rapid EBGP connection reset is disabled in a situation where the status of an interface used to establish an EBGP connection changes frequently. If the status of the interface becomes stable, run the **ebgp-interface-sensitive** command to enable rapid EBGP connection reset to implement rapid BGP network convergence.

----End

8.13.6 Enabling BGP Tracking

BGP tracking can be used to adjust the interval between peer unreachability discovery and connection interruption. This suppresses BGP peer relationship flapping caused by route flapping and improves BGP network stability.

Context

BGP can be configured to detect peer relationship status changes in order to implement rapid BGP convergence. BFD, however, needs to be configured on the entire network, and has poor extensibility. If BFD cannot be deployed on a device to detect BGP peer relationship status, BGP tracking can be enabled on the device to quickly detect link or peer unreachability, implementing rapid network convergence.

Perform the following steps on a BGP router.

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
bgp as-number
```

The BGP view is displayed.

Step 3 Run:

```
peer { group-name | ipv4-address } tracking [ delay delay-time ]
```

BGP tracking is enabled on the device to detect the status of a specified peer.

By default, BGP tracking is disabled.

ipv4-address specifies the address of a peer. *group-name* specifies the name of a peer group. BGP tracking configured on a peer takes precedence over BGP tracking configured on the peer group of this peer.

If *delay-time* is not specified, the default delay (0 seconds) is used. This means that a BGP device tears down the connection with a peer immediately after detecting the peer unreachable.

A proper *delay-time* value can ensure network stability.

- If an IBGP peer relationship is established based on an IGP route, the *delay-time* values set on BGP peers must be greater than the IGP route convergence time. Otherwise, if IGP route flapping occurs, the BGP peer relationship will be interrupted before network convergence is complete.

 **NOTE**

IGP GR is configured and a BGP peer relationship is established based on an IGP route. If a device becomes faulty and performs an active/standby switchover, the IGP will not delete routes received by the device. As a result, the BGP peer relationship will not be interrupted, even though BGP tracking does not take effect.

- If BGP peers have negotiated the GR capability and one of the peers performs an active/standby switchover, the *delay-time* values on the BGP peers must be greater than the GR

time. Otherwise, the BGP peer relationship will be interrupted before the GR time expires. As a result, GR becomes invalid.

----End

8.13.7 Checking the Configuration

After the BGP network convergence speed is adjusted, you can view information about BGP peers and peer groups.

Prerequisites

The configurations for adjusting the BGP network convergence speed are complete.

Procedure

- Run the **display bgp peer** [*verbose*] command to check information about BGP peers.
- Run the **display bgp group** [*group-name*] command to check information about BGP peer groups.

----End

8.14 Configuring BGP Route Dampening

BGP route dampening can be configured to suppress unstable routes.

Applicable Environment

The main cause of route instability is route flapping. A route is considered to be flapping when it repeatedly appears and then disappears in the routing table. BGP is generally applied to complex networks where routes change frequently. Frequent route flapping consumes lots of bandwidth and CPU resources and even seriously affects network operations.

BGP route dampening prevents frequent route flapping by using a penalty value to measure route stability. When a route flaps for the first time, a penalty value is assigned to the route. Later, each time the route flaps, the penalty value of the route increases by a specific value. The greater the penalty value, the less stable the route. If the penalty value of a route exceeds the pre-defined threshold, the route will not be advertised until the penalty value of the route reduces to the reuse threshold.

Route dampening applies only to EBGp routes. IBGP routes, however, cannot be dampened. Generally, IBGP routes include routes from the local AS, requiring that the forwarding tables be the same. In addition, IGP fast convergence aims to achieve information synchronization. If IBGP routes are dampened, dampening parameters vary on different devices, and the forwarding tables are inconsistent.

Pre-configuration Tasks

Before configuring BGP route dampening, complete the following task:

- [Configuring Basic BGP Functions](#)

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
bgp as-number
```

The BGP view is displayed.

Step 3 Run:

```
ipv4-family unicast
```

The IPv4 unicast address family view is displayed.

Step 4 Run:

```
dampening [ half-life-reach reuse suppress ceiling | route-policy route-policy-name ] *
```

BGP route dampening parameters are set.

NOTE

The **dampening** command takes effect only for EBGp routes.

When you configure BGP route dampening, the values of *reuse*, *suppress*, and *ceiling* should meet the relationship of *reuse*<*suppress*<*ceiling*.

If routes are differentiated based on policies and the **dampening** command is run to reference a route-policy, BGP can use different route dampening parameters to suppress different routes.

----End

Checking the Configuration

After BGP route dampening is configured, you can check whether the configuration is correct.

- Run the **display bgp routing-table flap-info [regular-expression as-regular-expression | as-path-filter as-path-filter-number | network-address [{ mask | mask-length } [longer-match]]]** command to check route flapping statistics.
- Run the **display bgp routing-table dampened** command to check dampened BGP routes.
- Run the **display bgp routing-table dampening parameter** command to check configured BGP route dampening parameters.

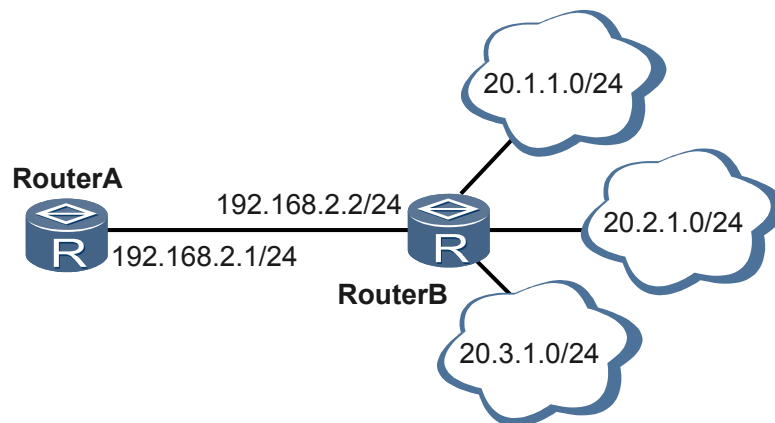
8.15 Configuring a BGP Device to Send a Default Route to Its Peer

After a BGP device is configured to send a default route to its peer, the BGP device sends a default route with the local address as the next-hop address to a specified peer, regardless of whether there are default routes in the local routing table. This greatly reduces the number of routes on the network.

Applicable Environment

The BGP routing table of a device on a medium or large BGP network contains a large number of routing entries. Storing the routing table consumes a large number of memory resources, and transmitting and processing routing information consume lots of network resources. If a device needs to send multiple routes to its peer, the device can be configured to send only a default route with the local address as the next-hop address to its peer, regardless of whether there are default routes in the local routing table. This greatly reduces the number of routes on the network and the consumption of memory resources on the peer and network resources.

Figure 8-2 Networking diagram for configuring a BGP device to send a default route to its peer



On the network shown in **Figure 8-2**, Router A and Router B have established a BGP peer relationship. Router B has imported routes to network segments 20.1.1.0/24, 20.2.1.0/24, and 20.3.1.0/24 to its BGP routing table. Router A needs to learn these routes from Router B. To reduce the consumption of memory resources of Router A and bandwidth used by Router B for sending routing information to Router A, configure Router B to send a default route to its peer (Router A) and use a routing policy to prevent all the routes to network segments 20.1.1.0/24, 20.2.1.0/24, and 20.3.1.0/24 from being sent to Router A. Then, Router A stores only one default route but can still send traffic to the three network segments.

Pre-configuration Tasks

Before configuring a BGP device to send a default route to its peer, complete the following task:

- **Configuring Basic BGP Functions**

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
bgp as-number
```

The BGP view is displayed.

Step 3 Run:

```
ipv4-family unicast
```


The IPv4 unicast address family view is displayed.

Step 4 Run:

```
peer { group-name | ipv4-address } default-route-advertise [ route-policy route-policy-name ] [ conditional-route-match-all { ipv4-address1 { mask1 | mask-length1 } } &<1-4> | conditional-route-match-any { ipv4-address2 { mask2 | mask-length2 } } &<1-4> ]
```

The device is configured to send a default route to a peer or a peer group.

If **route-policy route-policy-name** is set, the BGP device changes attributes of a default route based on the specified route policy.

If **conditional-route-match-all { ipv4-address1 { mask1 | mask-length1 } } &<1-4>** is set, the BGP device sends a default route to the peer only when all specified routes exist in the local routing table.

If **conditional-route-match-any { ipv4-address2 { mask2 | mask-length2 } } &<1-4>** is set, the local device sends a default route to the peer when one of the specified routes exists in the local routing table.

NOTE

After the **peer default-route-advertise** command is used on a device, the device sends a default route with the local address as the next-hop address to a specified peer, regardless of whether there is a default route in the routing table.

----End

Checking the Configuration

After a BGP device is configured to send a default route to a peer, you can check whether the configuration is correct.

- Run the **display bgp routing-table [ipv4-address [mask | mask-length]]** command on a peer to check information about a received BGP default route.

8.16 Configuring BGP Load Balancing

Configuring BGP load balancing better utilizes network resources and reduces network congestion.

Applicable Environment

On large networks, there may be multiple valid routes to the same destination. BGP, however, advertises only the optimal route to its peers. This may result in unbalanced traffic on different routes.

The following two methods can be used to address the problem of unbalanced traffic:

- Use BGP routing policies to allow traffic to be balanced. For example, use a routing policy to modify the Local_Pref, AS_Path, Origin, and Multi Exit Discriminator (MED) attributes of BGP routes to direct traffic to different forwarding paths for load balancing. For details on how to modify attributes of BGP routes, see [Configuring BGP Route Attributes](#).
- Use multiple paths for load balancing. In this method, multiple equal-cost routes need to be configured for traffic load balancing.

 **NOTE**

Equal-cost BGP routes can be generated for traffic load balancing only when the first 8 route attributes described in "Route Selection Policies for Load Balancing" in [BGP Features Supported by the AR150/200](#) are the same, and the AS-Path attributes are also the same.

Pre-configuration Tasks

Before configuring BGP load balancing, complete the following task:

- [Configuring Basic BGP Functions](#)

Data Preparation

To configure BGP load balancing, you need the following data.

No.	Data
1	Number of BGP routes to be used for load balancing
2	Number of EBGP and IBGP routes to be used for load balancing

Procedure

- Set the number of BGP routes to be used for load balancing.

Perform the following steps on a BGP router:

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
bgp as-number
```

The BGP view is displayed.

3. Run:

```
ipv4-family unicast
```

The IPv4 unicast address family view is displayed.

4. Run:

```
maximum load-balancing [ ebgp | ibgp ] number
```

The number of BGP routes to be used for load balancing is set.

By default, the number of BGP routes to be used for load balancing is 1, meaning that load balancing is not implemented.

- **ebgp** indicates that load balancing is implemented only among EBGP routes.
- **ibgp** indicates that load balancing is implemented only among IBGP routes.
- If neither **ebgp** nor **ibgp** is specified, both EBGP and IBGP routes participate in load balancing, and the number of EBGP routes to be used for load balancing is the same as the number of IBGP routes to be used for load balancing.

 **NOTE**

The **maximum load-balancing number** command cannot be configured together with the **maximum load-balancing ebgp number** or **maximum load-balancing ibgp number** command.

When routes with the same destination addresses carry out load balancing on the public network, the system determines the type of optimal routes first. If the optimal routes are IBGP routes, only IBGP routes carry out load balancing. If the optimal routes are EBGP routes, only EBGP routes carry out load balancing. This means that load balancing cannot be implemented among IBGP and EBGP routes with the same destination address.

5. (Optional) Run:

```
load-balancing as-path-ignore
```

The router is configured not to compare the AS-Path attributes of the routes to be used for load balancing.

By default, the router compares the AS-Path attributes of the routes to be used for load balancing.

 **NOTE**

- If there are multiple routes to the same destination but these routes pass through different ASs, load balancing cannot be implemented among these routes by default. To implement load balancing among these routes, run the **load-balancing as-path-ignore** command. After the **load-balancing as-path-ignore** command is run, the device no longer compares the AS-Path attributes of the routes to be used for load balancing. Therefore, exercise caution when using this command.
- The **load-balancing as-path-ignore** and **bestroute as-path-ignore** commands are mutually exclusive.
- Set the maximum number of EBGP and IBGP routes to be used for load balancing.

This configuration is used in a VPN where a CE is dual-homed to two PEs. When the CE and one PE belong to an AS and the CE and the other PE belong to a different AS, you can set the number of EBGP and IBGP routes to be used for load balancing. This allows VPN traffic to be balanced among EBGP and IBGP routes.

Perform the following steps on a BGP router:

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
bgp as-number
```

The BGP view is displayed.

3. Run:

```
ipv4-family vpn-instance vpn-instance-name
```

The BGP-VPN instance view is displayed.

4. Run:

```
maximum load-balancing eibgp number
```

The maximum number of EBGP and IBGP routes is set for load balancing.

By default, the maximum number of EBGP and IBGP routes to be used for load balancing is not set.

5. (Optional) Run:

load-balancing as-path-ignore

The router is configured not to compare the AS-Path attributes of the routes to be used for load balancing.

By default, the router compares the AS-Path attributes of the routes to be used for load balancing.

NOTE

- After the **load-balancing as-path-ignore** command is run, the router no longer compares the AS-Path attributes of the routes to be used for load balancing. Therefore, exercise caution when using this command.
- The **load-balancing as-path-ignore** and **bestroute as-path-ignore** commands are mutually exclusive.

---End

Checking the Configuration

After the BGP load balancing configurations are complete, you can run the following commands to check the configurations.

- Run the **display bgp routing-table** [*network* [{ *mask* | *mask-length* } [**longer-prefixes**]]] command to check routing information in a BGP routing table.
- Run the **display ip routing-table vpn-instance** *vpn-instance-name* [**verbose**] command to view the routing table of a VPN instance.

8.17 Configuring Path MTU Auto Discovery

Path MTU auto discovery allows BGP to discover the smallest MTU value on a path to ensure that BGP messages satisfy the path MTU requirement. This function improves transmission efficiency and BGP performance.

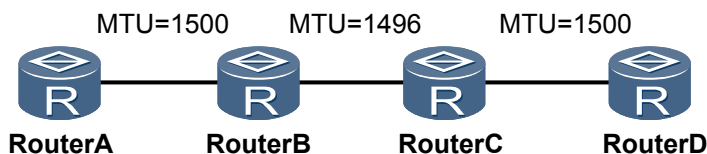
Applicable Environment

The link-layer MTUs of different networks that a communication path traverses vary from each other. The smallest MTU on the path is the most important factor that influences the communication between the two ends of the path and is called the path MTU.

The path MTU varies with the selected route and therefore may change. In addition, path MTUs in the inbound and outbound directions may be inconsistent. The path MTU auto discovery function is used to find the smallest MTU on the path from the source to the destination. The path MTU will be used as a basis for IP datagram fragmentation when TCP is used to transmit BGP messages.

As shown in [Figure 8-3](#), a BGP peer relationship is set up between Router A and Router D. BGP messages are encapsulated into TCP data packets for transmission. The default maximum segment size (MSS) is 536. Therefore, Router A sends TCP data packets of the default MSS of 536 to Router D. As a result, a lot of BGP messages are sliced and packed into different packets, and the number of ACK packets corresponding to these messages increases, leading to a low transmission efficiency. Path MTU auto discovery solves this problem. As shown in [Figure 8-3](#), the path MTU between Router A and Router D is 1496. To speed up BGP message transmission and improve BGP performance, configure path MTU auto discovery between Router A and Router D to allow BGP messages to be transmitted based on the MSS of 1496.

Figure 8-3 Networking diagram for path MTU auto discovery



Pre-configuration Tasks

Before configuring path MTU auto discovery, complete the following task:

- **Configuring Basic BGP Functions**

Data Preparation

To configure path MTU auto discovery, you need the following data.

No.	Data
1	(Optional) Aging time of the path MTU

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
bgp as-number
```

The BGP view is displayed.

Step 3 Run:

```
peer { group-name | ipv4-address } path-mtu auto-discovery
```

Path MTU auto discovery is enabled.

By default, path MTU auto discovery is disabled.

After the command is run, a BGP peer learns the path MTU, preventing BGP messages to be fragmented during transmission.

NOTE

The transmit and receive paths between two BGP peers may be different. Therefore, running this command on both ends is recommended. It makes both peers exchange messages based on the path MTU.

Step 4 Run:

```
quit
```

Return to the system view.

Step 5 Run:

```
tcp timer pathmtu-age age-time
```

The aging time is set for an IPv4 path MTU.

By default, the IPv4 path MTU aging time is set to 0 seconds. An IPv4 path MTU does not age.

The path MTUs of different routes may be different. The path MTU of hosts depends on the selected route and may change. If there are multiple routes between two communication hosts and the routes selected for packet transmission change frequently, the path MTU aging time needs to be configured. The system updates path MTUs based on the path MTU aging time, increasing the transmission efficiency.

----End

Checking the Configuration

After configuring path MTU auto discovery, you can run the following commands to check the previous configuration.

- Run the **display bgp peer [*ipv4-address*] verbose** command to check whether path MTU auto discovery has been successfully configured.

8.18 Configuring the BGP Next Hop Delayed Response

Configuring the BGP next hop delayed response can minimize traffic loss during route changes.

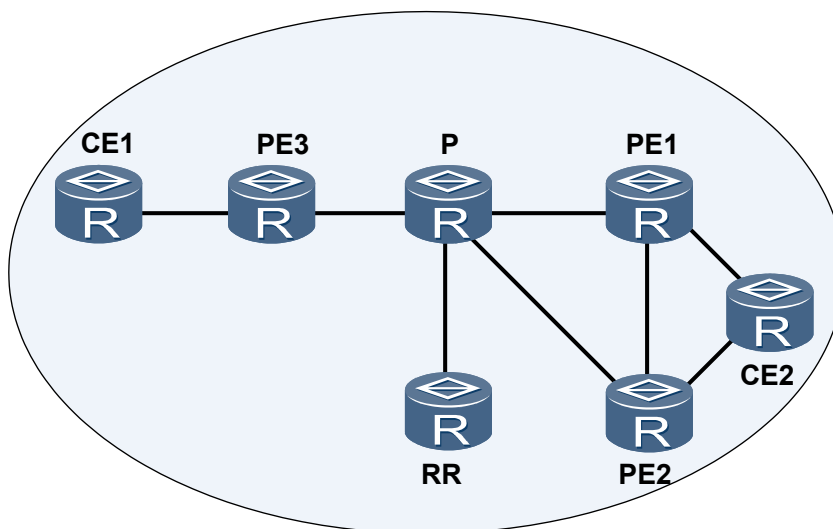
Context

Configuring the BGP next hop delayed response can speed up BGP route convergence and minimize traffic loss.

As shown in [Figure 8-4](#), PE1, PE2, and PE3 are the clients of the RR. CE2 is dual-homed to PE1 and PE2. PE1 and PE2 advertise their routes to CE2 to the RR. The RR advertises the route from PE1 to PE3. PE3 has a route to CE2 only and advertises this route to CE1. After the route exchange, CE1 and CE2 can communicate. If PE1 fails, PE3 detects that the next hop is unreachable and instructs CE1 to delete the route to CE2. Traffic is interrupted. After BGP route convergence is complete, the RR selects the route advertised by PE2 and sends a route update message to PE3. PE3 then advertises this route to CE1, and traffic forwarding is restored to the normal state. A high volume of traffic will be lost during traffic interruption because BGP route convergence is rather slow.

If the BGP next hop delayed response is enabled on PE3, PE3 does not reselect a route or instruct CE1 to delete the route to CE2 immediately after detecting that the route to PE1 is unreachable. After BGP convergence is complete, the RR selects the route advertised by PE2 and sends the route to PE3. PE3 then reselects a route and sends a route update message to CE1. Traffic forwarding is restored to the normal state. After the BGP next hop delayed response is enabled on PE3, PE3 does not need to delete the route or instruct CE1 to delete the route. This delayed response speeds up BGP route convergence and minimizes traffic loss.

Figure 8-4 Networking diagram for configuring the BGP next hop delayed response



The BGP next hop delayed response applies to a scenario where the next hop has multiple links to reach the same destination. If there is only one link between the next hop and the destination, configuring the BGP next hop delayed response may cause heavier traffic loss when the link fails because link switching is impossible.

Pre-configuration Tasks

Before configuring the BGP next hop delayed response, complete the following task:

- **Configuring Basic BGP Functions**

Data Preparation

To configure the BGP next hop delayed response, you need the following data.

No.	Data
1	Delay in responding to changes of the next hop

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
bgp as-number
```

The BGP view is displayed.

Step 3 Run:

```
nexthop recursive-lookup delay [ delay-time ]
```

A delay in responding to a next hop change is set.

The default delay time is 5 seconds.

 **NOTE**

BGP route convergence depends on IGP route convergence. If IGP route convergence is quick, the default delay time does not need to be changed. If IGP route convergence is slow, setting a delay time longer than IGP route convergence time is recommended.

----End

Checking the Configuration

After configuring the BGP next hop delayed response, you can run the following command to check the previous configuration.

- Run the **display current-configuration configuration bgp | include nexthop recursive-lookup delay** command to view information about the delay in responding to a next hop change.

8.19 Configuring BFD for BGP

BFD for BGP speeds up fault detection and therefore increases the route convergence speed.

Applicable Environment

As technologies develop, voice and video services are widely applied. These services are quite sensitive to the packet loss and delay. BGP periodically sends Keepalive packets to its peers to detect the status of its peers. The detection mechanism, however, takes more than one second. When the data transmission rate reaches the level of Gbit/s, such slow detection will cause a large amount of data to be lost. As a result, the requirement for high reliability of carrier-class networks cannot be met.

BFD for BGP can be used to reduce packet loss and delay. BFD for BGP detects faults on links between BGP peers within 50 milliseconds. The fast detection speed ensures fast BGP route convergence and minimizes traffic loss.

 **NOTE**

By default, a multi-hop BGP session is established between Huawei devices that set up an IBGP peer relationship. A BFD for IGP session and A BFD for IBGP session cannot be both set up between a Huawei device and a non-Huawei device that sets up a single-hop BGP session with its peer by default. In such a situation, setting up only A BFD for IGP session or A BFD for IBGP session between the Huawei and non-Huawei devices is recommended.

A BFD session currently does not detect route switching. If the change of bound peer IP address causes a route to switch to another link, the BFD session is negotiated again only when the original link fails.

Pre-configuration Tasks

Before configuring BFD for BGP, complete the following task:

- [Configuring Basic BGP Functions](#)

Data Preparation

To configure BFD for BGP, you need the following data.

No.	Data
1	IP address of the BGP peer or name of the peer group for which BFD needs to be configured
2	BFD parameters, including the minimum and maximum intervals for receiving BFD packets, Wait-to-Restore (WTR) time of a BFD session, and the detection multiplier
3	Name of the VPN instance for which BFD needs to be configured

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
bfd
```

BFD is enabled globally.

Step 3 Run:

```
quit
```

Return to the system view.

Step 4 Run:

```
bgp as-number
```

The BGP view is displayed.

Step 5 (Optional) Run:

```
ipv4-family vpn-instance vpn-instance-name
```

The BGP-VPN instance IPv4 address family view is displayed.

NOTE

BFD for BGP can be configured for the VPN in this view. To configure BFD for BGP for the public network, skip this step.

Step 6 Run:

```
peer { group-name | ipv4-address } bfd enable
```

BFD is enabled for the peer or peer group and a BFD session is established using default parameters.

After BFD is enabled for a peer group, BFD sessions will be created on the peers that belong to this peer group and are not configured with the **peer bfd block** command.

Step 7 (Optional) Run:

```
peer { group-name | ipv4-address } bfd { min-tx-interval min-tx-interval | min-rx-interval min-rx-interval | detect-multiplier multiplier | wtr wtr-value } *
```

BFD session parameters are modified.

 **NOTE**

The BFD parameters of peers take precedence over those of peer groups. If BFD parameters are configured on peers, they will be used in BFD session establishment.

The default interval for transmitting BFD packets and the default detection multiplier are recommended. When changing the default values, pay attention to the network status and the network reliability requirement. A short interval for transmitting BFD packets can be configured for a link that has a higher reliability requirement. A long interval for transmitting BFD packets can be configured for a link that has a lower reliability requirement.

 **NOTE**

There are three formulas: Actual interval for the local device to send BFD packets = max {Locally configured interval for transmitting BFD packets, Remotely configured interval for receiving BFD packets}, Actual interval for the local device to receive BFD packets = max {Remotely configured interval for transmitting BFD packets, Locally configured interval for receiving BFD packets}, and Local detection period = Actual interval for receiving BFD packets x Remotely configured BFD detection multiplier.

For example:

- On the local device, the configured interval for transmitting BFD packets is 200 ms, the interval for receiving BFD packets is 300 ms, and the detection multiplier is 4.
- On the peer device, the configured interval for transmitting BFD packets is 100 ms, the interval for receiving BFD packets is 600 ms, and the detection multiplier is 5.

Then:

- On the local device, the actual interval for transmitting BFD packets is 600 ms calculated by using the formula max {200 ms, 600 ms}; the interval for receiving BFD packets is 300 ms calculated by using the formula max {100 ms, 300 ms}; the detection period is 1500 ms calculated by multiplying 300 ms by 5.
- On the peer device, the actual interval for transmitting BFD packets is 300 ms calculated by using the formula max {100 ms, 300 ms}; the interval for receiving BFD packets is 600 ms calculated by using the formula max {200 ms, 600 ms}; the detection period is 2400 ms calculated by multiplying 600 ms by 4.

wtr *wtr-value* can be specified in the command to suppress frequent BFD and BGP session flapping caused by link flapping. If a BFD session over a link goes Down, it does not go Up immediately after the link recovers. Instead, the BFD session waits for the WTR timer to expire before going Up. If the link fails again before the WTR timer expires, BFD does not send a link fault message to BGP, and the BGP session status is stabilized.

The default value of *wtr-value* is 0, which means that the WTR timer will not be started.

Step 8 (Optional) Run:

```
peer ipv4-address bfd block
```

A peer is prevented from inheriting the BFD function of the peer group to which it belongs.

If a peer joins a peer group enabled with BFD, the peer inherits the BFD configuration of the group and creates a BFD session. To prevent the peer from inheriting the BFD function of the peer group, perform this step.

 **NOTE**

The **peer bfd block** command and the **peer bfd enable** command are mutually exclusive. After the **peer bfd block** command is run, the BFD session is automatically deleted.

----End

Checking the Configuration

After configuring BFD for BGP, you can run the following command to check the configurations.

- Run the **display bgp bfd session** { [**vpn4 vpn-instance** *vpn-instance-name*] **peer ipv4-address** | **all** } command to check information about the BFD session between BGP peers.

8.20 Configuring BGP GR

BGP GR can be configured to avoid traffic interruption due to protocol restart.

8.20.1 Establishing the Configuration Task

Before configuring BGP GR, familiarize yourself with the applicable environment, complete the pre-configuration tasks, and obtain the data required for the configuration. This will help you complete the configuration task quickly and efficiently.

Applicable Environment

BGP restart causes peer relationship reestablishment and traffic interruption. After GR is enabled, traffic interruption can be prevented in the event of BGP restart.

The following roles are involved in BGP GR:

- GR restarter: is a device that is restarted by the administrator or in the case of a failure. The GR restarter must be a GR-capable device.
- GR helper: is a neighbor of the GR restarter. The GR helper must also have the GR capability.

NOTE

The AR150/200 can function as only the Helper router, but cannot function as the Restarter router.

Pre-configuration Tasks

Before configuring BGP GR, complete the following task:

- [Configuring Basic BGP Functions](#)

Data Preparation

To configure BGP GR, you need the following data.

No.	Data
1	BGP AS number
2	Maximum period of time for reestablishing a BGP session
3	Period of time for waiting for End-Of-RIB messages

8.20.2 Enabling BGP GR

Enabling or disabling BGP GR may delete and re-establish all BGP sessions and instances.

Context

A GR-capable device can establish GR sessions with a GR-capable neighbor. By controlling the session negotiation mechanism of BGP, the GR restarter and the GR helper can understand each other's GR capability. When detecting the restart of the GR restarter, the GR helper does not delete the routing and forwarding entries related to the GR restarter, but waits to re-establish a BGP connection with the GR restarter. After establishing a new BGP connection, the GR restarter and the GR helper update BGP routes.

Perform the following steps on the router to be enabled with BGP GR:

Procedure

- Step 1** Run:
- ```
system-view
```
- The system view is displayed.
- Step 2** Run:
- ```
bgp as-number
```
- The BGP view is displayed.
- Step 3** Run:
- ```
graceful-restart
```
- BGP GR is enabled.
- By default, BGP GR is disabled.
- End

## 8.20.3 Configuring Parameters for a BGP GR Session

BGP GR session parameter values can be adjusted as needed, but default values are recommended. Changing the BGP restart period reestablishes BGP peer relationships.

### Context

GR time is the period of time during which the GR helper retains the forwarding information after having found the GR restarter Down. If the GR helper finds that the GR restarter goes Down, the GR helper keeps the topology information or routes learned from the GR restarter till the GR time expires.

Perform the following steps on the router to be enabled with BGP GR:

### Procedure

- Step 1** Run:
- ```
system-view
```
- The system view is displayed.

Step 2 Run:

```
bgp as-number
```

The BGP view is displayed.

Step 3 Run:

```
graceful-restart timer restart time
```

The maximum period of time used for reestablishing a BGP session is set.

The restart period of the router is the maximum waiting period from the time when the receiving speaker discovers that the peer restarts to the time when the BGP session is reestablished. By default, the restart period is 150 seconds.

 **NOTE**

Changing the BGP restart period reestablishes BGP peer relationships.

Step 4 Run:

```
graceful-restart timer wait-for-rib time
```

The length of time the restarting speaker and receiving speaker wait for End-of-RIB messages is set.

By default, the time for waiting for End-Of-RIB messages is 600s.

 **NOTE**

You can adjust BGP GR session parameter values as needed, but default values are recommended.

----End

8.20.4 Checking the Configuration

After BGP GR is configured, you can view the BGP GR status.

Prerequisites

The BGP GR configurations are complete.

Procedure

- Run the **display bgp peer verbose** command to check the BGP GR status.

----End

8.21 Configuring BGP Security

Authentication can be implemented during the establishment of a TCP connection to enhance BGP security.

8.21.1 Establishing the Configuration Task

Before configuring BGP security, familiarize yourself with the applicable environment, complete the pre-configuration tasks, and obtain the data required for the configuration. This will help you complete the configuration task quickly and efficiently.

Applicable Environment

MD5 authentication, keychain authentication, or GTSM can be configured on a BGP network to enhance BGP security.

- MD5 authentication

BGP uses TCP as the transport protocol and considers a packet valid as long as the source address, destination address, source port, destination port, and TCP sequence number of the packet are correct. Most parameters in a packet can be easily obtained by attackers. To protect BGP against attacks, MD5 authentication can be used during TCP connection establishment between BGP peers to reduce the possibility of attacks.

To prevent the MD5 password set on a BGP peer from being decrypted, you need to update the MD5 password periodically.

- Keychain authentication

A keychain consists of multiple authentication keys, each of which contains an ID and a password. Each key has a lifecycle. Based on the life cycle of a key, you can dynamically select different authentication keys from the keychain. After keychains with the same rules are configured on the two ends of a BGP connection, the keychains can dynamically select authentication keys to enhance BGP attack defense.

- GTSM

GTSM checks TTL values to defend against attacks. For example, an attacker forges BGP packets and keeps sending them to one router. After receiving these packets, the router identifies the destination of the packets. The forwarding plane of the router then directly sends the packets to the control plane for processing without checking the validity of the packets. As a result, the router is busy processing these "valid" packets, resulting in high CPU usage.

GTSM checks whether or not the TTL value in the IP header is within a specified range, protecting the router against attacks and improving system security.

 **NOTE**

- The AR150/200 supports GTSM.
- GTSM supports only unicast addresses; therefore, the GTSM function must be configured on all the routers configured with BGP.

Pre-configuration Tasks

Before configuring BGP security, complete the following task:

- [Configuring Basic BGP Functions](#)

Data Preparation

To configure BGP security, you need the following data.

No.	Data
1	Each router's peer address or peer group name
2	MD5 authentication password
3	Keychain authentication name

8.21.2 Configuring MD5 Authentication

In BGP, MD5 authentication sets an MD5 authentication password for a TCP connection, and is performed by TCP. If authentication fails, no TCP connection will be established.

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
bgp as-number
```

The BGP view is displayed.

Step 3 Run:

```
peer { ipv4-address | group-name } password { cipher cipher-password | simple  
simple-password }
```

An MD5 authentication password is set.

An MD5 authentication password can be set either in cipher or plain text.

- **cipher** *cipher-password* indicates that a password is recorded in cipher text. This means that a password is encrypted using a special algorithm and then recorded in a configuration file.
- **simple** *simple-password* indicates that a password is recorded in plain text. This means that a password is directly recorded in a configuration file.

NOTE

The **peer password** command run in the BGP view is also applicable to the BGP-VPNv4 address family view, because both BGP and BGP-VPNv4 use the same TCP connection.

An MD5 password cannot both start and end with symbols `$$@` because these symbols are used to identify types of old and new passwords during an upgrade.

----End

8.21.3 Configuring Keychain Authentication

Keychain authentication needs to be configured on two devices that establish a BGP peer relationship. The encryption algorithms and passwords for keychain authentication on both peers must be the same. This allows the peers to establish a TCP connection to exchange BGP packets.

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
bgp as-number
```

The BGP view is displayed.

Step 3 Run:

```
peer { ipv4-address | group-name } keychain keychain-name
```

Keychain authentication is configured.

Keychain authentication needs to be configured on two devices that establish a BGP peer relationship. The encryption algorithms and passwords for keychain authentication on both peers must be the same. This allows the peers to establish a TCP connection to exchange BGP packets.

Before configuring BGP keychain authentication, ensure that the keychain specified by *keychain-name* has been configured. Otherwise, no TCP connection can be set up between two BGP peers.

 **NOTE**

- The **peer keychain** command run in the BGP view is also applicable to the BGP-VPNv4 address family view, because both BGP and BGP-VPNv4 use the same TCP connection.
- BGP MD5 authentication and BGP keychain authentication are mutually exclusive.

----End

8.21.4 Configuring BGP GTSM

The GTSM function protects devices by checking whether the TTL value in the IP header is within a pre-defined range.

Procedure

- Adjust GTSM.

Perform the following steps on two devices that establish a BGP peer relationship:

1. Run:
system-view
The system view is displayed.
2. Run:
bgp as-number
The BGP view is displayed.
3. Run:
peer { group-name | ipv4-address } valid-ttl-hops [hops]

BGP GTSM is configured.

The valid TTL range of a checked packet is [255 - hops + 1, 255]. For example, the *hops* value is 1 for an EBGp direct route. This means that the valid TTL of the EBGp direct routes is 255. By default, the *hops* value is 255. This means that the valid TTL range is [1, 255].

 **NOTE**

- The **peer valid-ttl-hops** command run in the BGP view is also applicable to the BGP-VPNv4 address family view, because both BGP and BGP-VPNv4 use the same TCP connection.
- The configurations of GTSM and EBGp-MAX-HOP affect the TTL values of sent BGP packets, and the configurations of the two functions are mutually exclusive.

An interface board of a BGP device enabled with GTSM checks the TTL values in all received BGP packets. In actual networking, packets with the TTL values out of a specified range are either allowed to pass or discarded by GTSM. When the default action of GTSM is drop, an appropriate TTL value range needs to be set based on the

network topology. Packets with the TTL values out of the range will be discarded. This prevents bogus BGP packets from consuming CPU resources.

- Set the GTSM default action.

Perform the following steps on a GTSM-enabled router:

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
gtsm default-action { drop | pass }
```

The default action to be taken on the packets that do not match a GTSM policy is Drop.

By default, the action to be taken on the packets that do not match the GTSM policy is pass

 **NOTE**

If the default action is configured but no GTSM policy is configured, GTSM does not take effect.

- Configure the log function for dropped packets.

Perform the following steps on a GTSM-enabled router:

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
gtsm log drop-packet all
```

The log function is enabled on a specified board.

The log records information that GTSM drops packets, which helps locate faults.

----End

8.21.5 Checking the Configuration

After configuring BGP security, you can view authentication information about BGP peers.

Prerequisites

The BGP security configurations are complete.

Procedure

- Run the **display bgp peer [ipv4-address] verbose** command to check detailed information about MD5 and keychain authentication on BGP peers.
- Run the **display bgp peer verbose** command to check whether the GTSM function is enabled on BGP peers and check the configured maximum valid TTL value.

- Run the **display gtsm statistics all** command to check GTSM statistics on all boards, including the total number of packets, the number of passed packets, and the number of dropped packets.

---End

8.22 Maintaining BGP

Maintaining BGP involves resetting a BGP connection and clearing BGP statistics.

8.22.1 Resetting BGP Connections

You can also reset BGP in GR mode. Resetting a BGP connection will interrupt the peer relationship.

Context



CAUTION

The BGP peer relationship is interrupted after you reset BGP connections with the **reset bgp** command. Exercise caution when running this command.

When the BGP routing policy on the router that does not support Route-refresh changes, you need to reset BGP connections to validate the configuration. To reset BGP connections, run the following **reset** commands in the user view.

Procedure

- To validate the new configurations, run the **reset bgp all** command in the user view to reset all BGP connections.
- To validate the new configurations, run the **reset bgp as-number** command in the user view to reset the BGP connection between the specified AS.
- To validate the new configurations, run the **reset bgp ipv4-address** command in the user view to reset the BGP connection between a specified peer.
- To validate the new configurations, run the **reset bgp external** command in the user view to reset all the EBGP connections.
- To validate the new configurations, run the **reset bgp group group-name** command in the user view to reset the BGP connection with the specified peer-groups.
- To validate the new configurations, run the **reset bgp internal** command in the user view to reset all IBGP connections.

---End

8.22.2 Clearing BGP Information

This section describes how to clear the statistics of BGP accounting, flapped routes, and suppressed routes.

Context



CAUTION

BGP statistics cannot be restored after being cleared. Exercise caution when running this command.

Procedure

- Run the **reset bgp flap-info** [**regexp as-path-regexp** | **as-path-filter** { *as-path-filter-number* | *as-path-filter-name* } | *ipv4-address* [*mask* | *mask-length*]] command in the user view to clear the statistics of flapped routes.
- Run the **reset bgp dampening** [*ipv4-address* [*mask* | *mask-length*]] command in the user view to clear the dampened routes and advertise the suppressed routes.
- Run the **reset bgp ipv4-address flap-info** command in the user view to clear the statistics of route flapping.

---End

8.23 Configuration Examples

BGP configuration examples explain networking requirements, networking diagram, configuration notes, configuration roadmap, and configuration procedure.

8.23.1 Example for Configuring Basic BGP Functions

Before building BGP networks, you need to configure basic BGP functions.

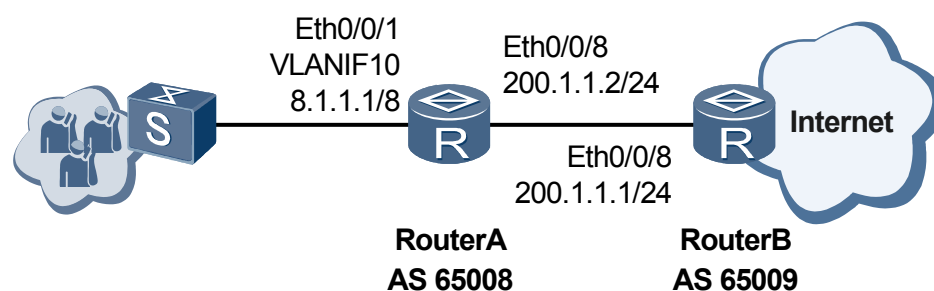
Networking Requirements

As shown in [Figure 8-5](#), there are two ASs: AS 65008 and AS 65009. RouterA belongs to AS 65008 and RouterB belongs to AS 65009. RouterA and RouterB establish an EBGP connection.

NOTE

AR150/200 can function only as RouterA in this scenario.

Figure 8-5 Networking diagram for configuring basic BGP functions



Configuration Roadmap

The configuration roadmap is as follows:

1. Configure an EBGP connection between RouterA and RouterB.
2. Run the **network** command on RouterA to advertise routes, and then check the routing tables of RouterA and RouterB.

Data Preparation

To complete the configuration, you need the following data:

- RouterA's router ID 1.1.1.1 and AS ID 65008
- RouterB's router ID 2.2.2.2 and AS ID 65009

Procedure

Step 1 Assign IP addresses to interfaces. The configuration procedure is not mentioned here.

Step 2 Configure EBGP.

Configure RouterA.

```
[RouterA] bgp 65008  
[RouterA-bgp] router-id 1.1.1.1  
[RouterA-bgp] peer 200.1.1.1 as-number 65009
```

Configure RouterB.

```
[RouterB] bgp 65009  
[RouterB-bgp] router-id 2.2.2.2  
[RouterB-bgp] peer 200.1.1.2 as-number 65008
```

View the status of BGP peers.

```
[RouterB-bgp] quit  
[RouterB] display bgp peer  
  
BGP local router ID : 2.2.2.2  
Local AS number : 65009  
Total number of peers : 1 Peers in established state : 1  
  
Peer V AS MsgRcvd MsgSent OutQ Up/Down State  
PrefRcv  
200.1.1.2 4 65008 38 38 0 00:35:56 Established  
1
```

An EBGP connection between Router B and RouterA has been established.

Step 3 Configure RouterA to advertise the route 8.0.0.0/8.

Configure RouterA to advertise routes.

```
[RouterA-bgp] ipv4-family unicast  
[RouterA-bgp-af-ipv4] network 8.0.0.0 255.0.0.0
```

Check the routing table of RouterA.

```
[RouterA-bgp-af-ipv4] quit  
[RouterA-bgp] quit  
[RouterA] display bgp routing-table  
  
BGP Local router ID is 1.1.1.1
```

```
Status codes: * - valid, > - best, d - damped,
              h - history, i - internal, s - suppressed, S - Stale
              Origin : i - IGP, e - EGP, ? - incomplete
```

```
Total Number of Routes: 1
      Network          NextHop          MED          LocPrf      PrefVal Path/Ogn
* >  8.0.0.0          0.0.0.0          0            0           0       i
```

Check the routing table of RouterB.

```
[RouterB] display bgp routing-table
```

```
BGP Local router ID is 2.2.2.2
Status codes: * - valid, > - best, d - damped,
              h - history, i - internal, s - suppressed, S - Stale
              Origin : i - IGP, e - EGP, ? - incomplete
```

```
Total Number of Routes: 1
      Network          NextHop          MED          LocPrf      PrefVal Path/Ogn
* >  8.0.0.0          200.1.1.2        0            0           0       65008i
```

----End

Configuration Files

- Configuration file of RouterA

```
#
sysname RouterA
#
vlan batch 10
#
interface Vlanif10
ip address 8.1.1.1 255.0.0.0
#
interface Ethernet0/0/1
port link-type trunk
port trunk allow-pass vlan 10
#
interface Ethernet0/0/8
ip address 200.1.1.2 255.255.255.0
#
bgp 65008
router-id 1.1.1.1
peer 200.1.1.1 as-number 65009
#
ipv4-family unicast
undo synchronization
network 8.0.0.0
peer 200.1.1.1 enable
#
return
```

- Configuration file of RouterB

```
#
sysname RouterB
#
interface Ethernet0/0/8
ip address 200.1.1.1 255.255.255.0
#
bgp 65009
router-id 2.2.2.2
peer 200.1.1.2 as-number 65008
#
ipv4-family unicast
```

```
undo synchronization
peer 200.1.1.2 enable
#
return
```

8.23.2 Example for Configuring BGP Community Attributes for Routes

Community attributes can be used to control BGP route selection.

Networking Requirements

Enterprises A, B, and C belong to different ASs. Enterprise B's network communicates with the networks of the other two enterprises using EBGP. Due to the competition relationship with enterprise C, enterprise A hopes that the routes it advertises to enterprise B are transmitted only in enterprise B, not to enterprise C. Community attributes can be configured for routes to be advertised by enterprise A to enterprise B in order to address this problem.

On the network shown in [Figure 8-6](#), Router B establishes EBGP connections with Routers A and C. To ensure that routes imported by Router A are transmitted only AS 20 after being advertised to Router B, you can configure the No_Export community attribute for BGP routes to be advertised by Router A so that the BGP's are sent only to AS 20.


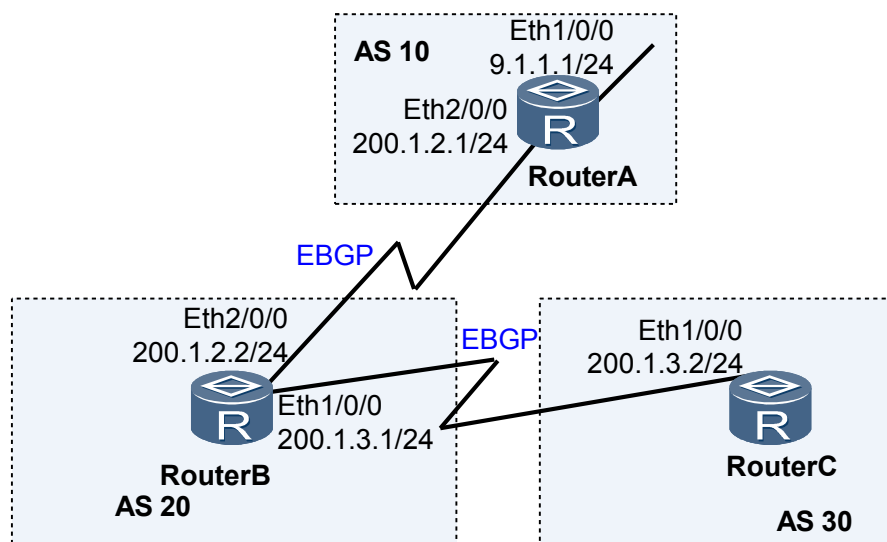
 **NOTE**
AR150/200 is RouterC.

Figure 8-6 Networking diagram of configuring the BGP community



Configuration Roadmap

The configuration roadmap is as follows:

1. Establish EBGP connections between Routers A and B and between Routers B and C so that the ASs can communicate with each other.

2. Use a routing policy on Router A to configure the No_Export community attribute for BGP routes to be advertised by Router A to Router B so that the routes are transmitted only in AS 20 and not to other ASs.

Data Preparation

To complete the configuration, you need the following data:

- Router ID 1.1.1.1 and AS number 10 of Router A
- Router ID 2.2.2.2 and AS number 20 of Router B
- Router ID 3.3.3.3 and AS number 30 of Router C

Procedure

Step 1 Configure an IP address for each interface. The configuration details are not provided here.

Step 2 Establish EBGP connections.

Configure Router A.

```
[RouterA] bgp 10
[RouterA-bgp] router-id 1.1.1.1
[RouterA-bgp] peer 200.1.2.2 as-number 20
[RouterA-bgp] ipv4-family unicast
[RouterA-bgp-af-ipv4] network 9.1.1.0 255.255.255.0
[RouterA-bgp-af-ipv4] quit
[RouterA-bgp] quit
```

Configure Router B.

```
[RouterB] bgp 20
[RouterB-bgp] router-id 2.2.2.2
[RouterB-bgp] peer 200.1.2.1 as-number 10
[RouterB-bgp] peer 200.1.3.2 as-number 30
[RouterB-bgp] quit
```

Configure Router C.

```
[RouterC] bgp 30
[RouterC-bgp] router-id 3.3.3.3
[RouterC-bgp] peer 200.1.3.1 as-number 20
[RouterC-bgp] quit
```

On Router B, view detailed information about route 9.1.1.0/24.

```
[RouterB] display bgp routing-table 9.1.1.0

BGP local router ID : 2.2.2.2
Local AS number : 20
Paths: 1 available, 1 best, 1 select
BGP routing table entry information of 9.1.1.0/24:
From: 200.1.2.1 (1.1.1.1)
Route Duration: 00h00m42s
Direct Out-interface: Ethernet2/0/0
Original nexthop: 200.1.2.1
Qos information : 0x0
AS-path 10, origin igp, MED 0, pref-val 0, valid, external, best, select, active,
pre 255
Advertised to such 2 peers:
    200.1.2.1
    200.1.3.2
```

The preceding command output shows that Router B advertises the received BGP route to Router C in AS 30.

View the BGP routing table of Router C.

```
[RouterC] display bgp routing-table

BGP Local router ID is 3.3.3.3
Status codes: * - valid, > - best, d - damped,
              h - history, i - internal, s - suppressed, S - Stale
              Origin : i - IGP, e - EGP, ? - incomplete

Total Number of Routes: 1
   Network                NextHop           MED           LocPrf     PrefVal Path/Ogn
*>  9.1.1.0/24            200.1.3.1                0           20 10i
```

The preceding command output shows that Router C has learned route 9.1.1.0/24 from Router B.

Step 3 Configure a BGP community attribute.

Configure a routing policy on Router A to prevent BGP routes to be advertised by Router A to Router B from being advertised to any other AS.

```
[RouterA] route-policy comm_policy permit node 10
[RouterA-route-policy] apply community no-export
[RouterA-route-policy] quit
```

Apply the routing policy.

```
[RouterA] bgp 10
[RouterA-bgp] ipv4-family unicast
[RouterA-bgp-af-ipv4] peer 200.1.2.2 route-policy comm_policy export
[RouterA-bgp-af-ipv4] peer 200.1.2.2 advertise-community
```

On Router B, view detailed information about route 9.1.1.0/24.

```
[RouterB] display bgp routing-table 9.1.1.0

BGP local router ID : 2.2.2.2
Local AS number : 20
Paths: 1 available, 1 best, 1 select
BGP routing table entry information of 9.1.1.0/24:
From: 200.1.2.1 (1.1.1.1)
Route Duration: 00h00m09s
Direct Out-interface: Ethernet2/0/0
Original nexthop: 200.1.2.1
Qos information : 0x0
Community: no-export
AS-path 10, origin igp, MED 0, pref-val 0, valid, external, best, select, active,
pre 255
Not advertised to any peer yet
```

The preceding command output shows that route 9.1.1.0/24 carries the configured community attribute and Router B does not advertise this route to any other AS.

----End

Configuration Files

- Configuration file of Router A

```
#
sysname RouterA
#
interface Ethernet1/0/0
ip address 9.1.1.1 255.255.255.0
#
interface Ethernet2/0/0
```



```
    ip address 200.1.2.1 255.255.255.0
#
bgp 10
  router-id 1.1.1.1
  peer 200.1.2.2 as-number 20
#
  ipv4-family unicast
    undo synchronization
    network 9.1.1.0 255.255.255.0
    peer 200.1.2.2 enable
    peer 200.1.2.2 route-policy comm_policy export
    peer 200.1.2.2 advertise-community
#
  route-policy comm_policy permit node 10
  apply community no-export
#
return
```

● Configuration file of Router B

```
#
sysname RouterB
#
interface Ethernet2/0/0
  ip address 200.1.2.2 255.255.255.0
#
interface Ethernet3/0/0
  ip address 200.1.3.1 255.255.255.0
#
bgp 20
  router-id 2.2.2.2
  peer 200.1.2.1 as-number 10
  peer 200.1.3.2 as-number 30
#
  ipv4-family unicast
    undo synchronization
    peer 200.1.2.1 enable
    peer 200.1.3.2 enable
#
return
```

● Configuration file of Router C

```
#
sysname RouterC
#
interface Ethernet1/0/0
  ip address 200.1.3.2 255.255.255.0
#
bgp 30
  router-id 3.3.3.3
  peer 200.1.3.1 as-number 20
#
  ipv4-family unicast
    undo synchronization
    peer 200.1.3.1 enable
#
return
```

9 BGP4+ Configuration

About This Chapter

BGP4+, which is applicable to the large-scale IPv6 network with a complicated structure, is used between ASs to transmit routing information.

[9.1 BGP4+ Overview](#)

BGP4+ is mainly used to control route transmission and select optimal routes.

[9.2 BGP4+ Features Supported by the AR150/200](#)

The system supports various BGP4+ features, including load balancing, route aggregation, route dampening, community, route reflector, confederation, BGP4+ accounting, 6PE, BFD for BGP4+, BGP4+ GR, and BGP4+ NSR.

[9.3 Configuring Basic BGP4+ Functions](#)

Before building BGP4+ networks, you need to configure basic BGP4+ functions.

[9.4 Configuring BGP4+ Route Attributes](#)

BGP4+ has many route attributes. By configuring these attributes, you can change BGP4+ routing policies.

[9.5 Controlling the Advertising and Receiving of BGP4+ Routing Information](#)

BGP4+ can perform routing policies on or filter only the routes to be advertised to a certain peer.

[9.6 Configuring Parameters of a Connection Between BGP4+ Peers](#)

By setting parameters of a connection between BGP4+ peers, you can adjust and optimize the BGP4+ network performance.

[9.7 Configuring BGP4+ Tracking](#)

On a network where BFD is unsuitable to deploy, you can configure BGP4+ tracking to implement the fast convergence of IBGP routes.

[9.8 Configuring BGP4+ Route Dampening](#)

By configuring BGP4+ route dampening, you can suppress unstable BGP4+ routes.

[9.9 Configuring BGP4+ Load Balancing](#)

Configuring BGP4+ load balancing better utilizes network resources and reduces network congestion.

[9.10 Configuring a BGP4+ Peer Group](#)

By configuring a BGP4+ peer group, you can simplify the management of routing policies, and thus improve the efficiency of route advertisement.

[9.11 Configuring a BGP4+ Route Reflector](#)

By configuring a BGP4+ route reflector, you can solve the problem of establishing fully meshed connections between multiple IBGP peers.

[9.12 Configuring a BGP4+ Confederation](#)

On a large-scale BGP4+ network, configuring a BGP4+ confederation can simplify the management of routing policies and improve the efficiency of route advertisement.

[9.13 Configuring BGP4+ Security](#)

To improve BGP4+ security, you can perform TCP connection authentication.

[9.14 Maintaining BGP4+](#)

Maintaining BGP4+ involves resetting a BGP4+ connection and clearing BGP4+ statistics.

[9.15 Configuration Examples](#)

BGP4+ configuration examples explain networking requirements, networking diagram, configuration notes, configuration roadmap, and configuration procedure.

9.1 BGP4+ Overview

BGP4+ is mainly used to control route transmission and select optimal routes.

BGP4+ is a dynamic routing protocol used between Autonomous Systems (ASs), and it is an extension of BGP.

The traditional BGP4 can manage only the IPv4 routing information. For other network layer protocols such as IPv6, the traditional BGP4 has a limited capability to transmit routing information.

The IETF introduces BGP4+ as a supplement to BGP4 to support multiple network layer protocols. The RFC for BGP4+ is RFC 2858 (Multiprotocol Extensions for BGP4).

To support IPv6, BGP4 needs to reflect the IPv6 protocol information to the Network Layer Reachable Information (NLRI) attribute and the Next_Hop attribute.

BGP4+ introduces two NLRI attributes:

- Multiprotocol Reachable NLRI (MP_REACH_NLRI): advertises the reachable routes and the next hop information.
- Multiprotocol Unreachable NLRI (MP_UNREACH_NLRI): withdraws the unreachable routes.

The Next_Hop attribute of BGP4+ is in the format of an IPv6 address. It can be an IPv6 global unicast address or the link-local address of the next hop.

BGP4+ can be applied to an IPv6 network by using the BGP attribute of multiple protocol extension. The message and routing mechanisms of BGP remain unaltered.

9.2 BGP4+ Features Supported by the AR150/200

The system supports various BGP4+ features, including load balancing, route aggregation, route dampening, community, route reflector, confederation, BGP4+ accounting, 6PE, BFD for BGP4+, BGP4+ GR, and BGP4+ NSR.

Most of BGP4+ features supported by the AR150/200 are similar to those of BGP supported by the AR150/200. For details, refer to the chapter "BGP Configuration".

BGP4+ does not support summary automatic and MP-BGP.

NOTE

The BGP4+ function is used with a license. To use the BGP4+ function, apply for and purchase the following license from the Huawei local office:

- AR150&200 Value-Added Data Package

9.3 Configuring Basic BGP4+ Functions

Before building BGP4+ networks, you need to configure basic BGP4+ functions.

9.3.1 Establishing the Configuration Task

Before configuring basic BGP4+ functions, familiarize yourself with the applicable environment, complete the pre-configuration tasks, and obtain the required data. This can help you complete the configuration task quickly and accurately.

Applicable Environment

BGP4+ is configured in an IPv6 network.

Pre-configuration Tasks

Before configuring basic BGP4+ functions, complete the following tasks:

- Enabling IPv6
- Configuring link layer protocol parameters and IPv6 addresses for interfaces to make link layers of the interfaces Up

Data Preparation

To configure BGP4+, you need the following data.

No.	Data
1	Local AS number and Router ID
2	IPv6 address and AS number of the peer
3	(Optional) Interfaces that set up the BGP4+ session

9.3.2 Starting a BGP Process

Starting a BGP4+ process is a prerequisite for configuring basic BGP4+ functions. When starting a BGP4+ process, you need to specify the number of the AS that the device belongs to.

Context

Do as follows on the router on which the BGP4+ connection needs to be set up:

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
bgp as-number
```

BGP is enabled (the local AS number is specified) and the BGP view is displayed.

Step 3 (Optional) Run:

```
router-id ipv4-address
```

The router ID is set.

Setting or changing the router ID of BGP resets the BGP peer relationship between routers.

 **TIP**

- To enhance the network reliability, you can manually configure the address of a loopback interface as the router ID. If the router ID is not set, BGP uses the router ID in the system view. To select the router ID in the system view, refer to the *Command Reference*.
- If no interface of a router is configured with an IPv4 address, you must set a router ID for the router.

---End

9.3.3 Configuring an IPv6 Peer

Devices can exchange BGP4+ routing information only after BGP4+ peers are configured and the BGP4+ peer relationship is established.

Procedure

- Configuring an IBGP Peer

Do as follows on the router on which the IBGP connection needs to be set up:

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
bgp as-number
```

The BGP view is displayed.

3. Run:

```
peer { ipv6-address | group-name } as-number as-number
```

The peer address and the AS where the peer resides are configured.

The AS number of the specified peer must be the same as the local AS number.

When the IPv6 address of a specified peer is a loopback address or a sub-interface address, you need to perform [Configuring the Local Interfaces Used for BGP4+ Connections](#) to ensure the establishment of the peer.

4. (Optional) Run:

```
peer { ipv6-address | group-name } listen-only
```

A peer (group) is configured only to listen to connection requests, but not to send connection requests.

After this command is used, the existing peer relationship is interrupted. The peer on which this command is used waits for the connection request from its peer to reestablish the neighbor relationship. This configuration can prevent the conflict of sending connection requests.

 **NOTE**

This command can be used on only one of two peers. If this command is used on the two peers, the connection between the two peers cannot be established.

5. Run:

```
ipv6-family [ unicast ]
```

The BGP IPv6 unicast address family view is displayed.

6. Run:

```
peer { ipv6-address | group-name } enable
```

The IPv6 peers are enabled.

After configuring the BGP4+ peers in the BGP view, you need to enable these peers in the BGP IPv6 unicast address family view.

● Configuring an EBGP Peer

Do as follows on the router on which the EBGP connection needs to be set up:

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
bgp as-number
```

The BGP view is displayed.

3. Run:

```
peer { ipv6-address | group-name } as-number as-number
```

The IP address and the AS number of a specified BGP peer are specified.

The AS number of the specified BGP peer should be different from the local AS number.

If the IP address of the specified peer is that of a loopback interface on the reachable peer or that of a sub-interface on the directly connected peer, you need to complete the task of [Configuring the Local Interfaces Used for BGP4+ Connections](#) to ensure that the peer is correctly established.

4. Run:

```
peer { ipv6-address | group-name } ebgp-max-hop [ hop-count ]
```

The maximum number of hops in the EBGP connections is set.

Usually, a direct physical link should be available between the EBGP peers. If this requirement cannot be met, you can use the **peer ebgp-max-hop** command to configure the EBGP peers to establish the TCP connections through multiple hops.

 NOTE

When establishing the EBGP connection through loopback interfaces, you must use the **peer ebgp-max-hop** command specifying that *hop-count* is greater than or equal to 2. Otherwise, BGP cannot set up the EBGP connection with the peer.

5. (Optional) Run:

```
peer { ipv6-address | group-name } listen-only
```

The peer or peer group is configured only to listen to connection requests, but not to send any connection request.

After this command is used, the existing peer relationship is removed. The peer on which this command is used reestablishes the peer relationship after receiving the connection request from its peer. After this configuration is done, the conflict of connection requests is avoided.

 **NOTE**

This command can be used on only one of two peers. If this command is used on the two peers, the connection between the two peers cannot be established.

6. Run:

```
ipv6-family [ unicast ]
```

The BGP IPv6 unicast address family view is displayed.

7. Run:

```
peer { ipv6-address | group-name } enable
```

An IPv6 peer is enabled.

After configuring a BGP4+ peer in the BGP view, enable the peer in the BGP IPv6 unicast address family view.

----End

9.3.4 (Optional) Configuring the Local Interfaces Used for BGP4+ Connections

When establishing BGP4+ peer relationship between two devices through various links, you need to specify the local interface during the setup of a BGP4+ session on the devices.

Context

Do as follows on the BGP4+ router:

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
bgp as-number
```

The BGP view is displayed.

Step 3 Run:

```
peer { ipv6-address | group-name } connect-interface interface-type interface-  
number [ ipv6-source-address ]
```

The source interface and source address used to set up a TCP connection are specified.

Usually, BGP4+ uses the physical interface that is directly connected with the peer as the session interface used for the TCP connection.

To increase the reliability and stability of the BGP4+ connections, configure the local interface used for the BGP4+ connection as the loopback interface. In this way, when there are redundant links on the network, the BGP4+ connections are not interrupted due to the failure of a certain interface or a link.

 **NOTE**

When establishing BGP4+ peer relationship between two devices through various links, specify the local interface during the setup of a BGP4+ session on the devices by using the **peer connect-interface** command is recommended.

----End

9.3.5 Checking the Configuration

After basic BGP4+ functions are configured, you can check BGP4+ peer information.

Prerequisites

The configurations for basic BGP4+ functions are complete.

Procedure

- Run the **display bgp ipv6 peer ipv6-address { log-info | verbose }** command to check information about the BGP4+ peers.

----End

9.4 Configuring BGP4+ Route Attributes

BGP4+ has many route attributes. By configuring these attributes, you can change BGP4+ routing policies.

9.4.1 Establishing the Configuration Task

Before controlling BGP4+ route selection, familiarize yourself with the applicable environment, complete the pre-configuration tasks, and obtain the required data. This can help you complete the configuration task quickly and accurately.

Applicable Environment

You can change the BGP4+ routing policies by configuring the route attributes.

- BGP4+ priority
After the BGP4+ priority is configured, Route Management (RM) is affected in routing between BGP4+ and the other routing protocols.
- Preferred value of BGP4+ routing information
After the preferred value of BGP4+ routing information is configured, the route with the greatest preferred value is selected when multiple routes to the same destination exist in the BGP4+ routing table.
- Local_Pref attribute
The function of the Local_Pref attribute is similar to that of the preferred value of BGP4+ routing information. The preferred value of BGP4+ routing information takes precedence over the Local_Pref attribute.
- MED attribute
After the MED attribute is configured, EBGP peers select the route with the smallest MED value when the traffic enters an AS.

- Next_Hop attribute
A route with an unreachable next hop is ignored.
- Community attribute
The community attribute can simplify the management of routing policies. The management range of the community attribute is wider than that of the peer group. The community attribute can control the routing policies of multiple BGP4+ routers.
- AS_Path attribute
After the AS_Path attribute is configured, the route with a shorter AS path is selected.

Pre-configuration Tasks

Before configuring BGP4+ route attributes, complete the following tasks:

- [Configuring Basic BGP4+ Functions](#)

Data Preparation

To configure BGP4+ route attributes, you need the following data.

No.	Data
1	AS number
2	Protocol priority
3	Local_Pref
4	MED
5	Name of the routing policy for using the community attribute

9.4.2 Configuring the BGP4+ Preference

Setting the BGP4+ preference can affect route selection between BGP4+ and another routing protocol.

Context

Do as follows on the BGP4+ router:

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
bgp as-number
```

The BGP view is displayed.

Step 3 Run:

```
ipv6-family [ unicast ]
```

The BGP IPv6 unicast address family view is displayed.

Step 4 Run:

```
preference { external internal local | route-policy route-policy-name }
```

The BGP4+ preference is set.

 **NOTE**

Using **peer route-policy** command to configure the preference of the BGP protocol on the peers is not currently supported.

----End

9.4.3 Configuring BGP4+ Preferred Value for Routing Information

After the preferred value is configured for routing information, the route with the largest preferred value is selected when multiple routes to the same destination exist in the BGP4+ routing table.

Context

Do as follows on the BGP4+ router:

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
bgp as-number
```

The BGP view is displayed.

Step 3 Run:

```
ipv6-family [ unicast ]
```

The BGP IPv6 unicast address family view is displayed.

Step 4 Run:

```
peer { group-name | ipv6-address } preferred-value value
```

The preferred value of a peer is configured.

By default, the preferred value of the route learned from a neighbor is 0.

----End

9.4.4 Configuring the Default Local_Pref Attribute of the Local Router

The Local_Pref attribute is used to determine the optimal route for the traffic that leaves an AS. When a BGP4+ router obtains multiple routes to the same destination address but with different next hops from different IBGP peers, the route with the largest Local_Pref value is selected.

Context

Do as follows on the BGP4+ router:

Procedure

- Step 1** Run:
`system-view`
The system view is displayed.
- Step 2** Run:
`bgp as-number`
The BGP view is displayed.
- Step 3** Run:
`ipv6-family [unicast]`
The BGP IPv6 unicast address family view is displayed.
- Step 4** Run:
`default local-preference preference`
The default Local_Pref of the local router is configured.
- End

9.4.5 Configuring the MED Attribute

The MED attribute serves as the metric used by an IGP. After MED attributes are set, EBGP peers select the route with the smallest MED value for the traffic that enters an AS.

Context

Do as follows on the BGP4+ router:

Procedure

- Step 1** Run:
`system-view`
The system view is displayed.
- Step 2** Run:
`bgp as-number`
The BGP view is displayed.
- Step 3** Run:
`ipv6-family [unicast]`
The BGP IPv6 unicast address family view is displayed.
- Step 4** Run the following commands to configure the BGP4+ MED attribute as required:
- Run:
`default med med`

The default MED attribute is configured.

- Run:

```
compare-different-as-med
```

The MED values from different ASs are compared.

- Run:

```
deterministic-med
```

Deterministic-MED is enabled.

If this command is not configured, when an optimal route is to be selected from among routes which are received from different ASs and which carry the same prefix, the sequence in which routes are received is relevant to the result of route selection. After the command is configured, however, when an optimal route is to be selected from among routes which are received from different ASs and which carry the same prefix, routes are first grouped according to the leftmost AS in the AS_Path. Routes with the same leftmost AS are grouped together, and after comparison, an optimal route is selected for the group. The group optimal route is then compared with optimal routes from other groups to determine the final optimal route. This mode of route selection ensures that the sequence in which routes are received is no longer relevant to the result of route selection.

- Run:

```
bestroute med-none-as-maximum
```

The maximum MED value is used when the current MED is not available.

- Run:

```
bestroute med-confederation
```

The MED values of routes advertised in the local confederation are compared.

The commands in Step 4 can be used regardless of the order.

----End

9.4.6 Configuring the Next_Hop Attribute

By setting the Next_Hop attribute, you can flexibly control BGP4+ route selection.

Procedure

- Modifying the Next Hop When Advertising a Route to an IBGP Peer

Do as follows on the IBGP router:

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
bgp as-number
```

The BGP view is displayed.

3. Run:

```
ipv6-family [ unicast ]
```

The BGP IPv6 unicast address family view is displayed.

4. Run:

```
peer ipv6-address next-hop-local
```

The local address is configured as the next hop when routes are advertised.

In some networking environments, to ensure that the IBGP neighbors find the correct next hop, configure the next hop address as its own address when routes are advertised to the IBGP peers.

 **NOTE**

If BGP load balancing is configured, the local router changes the next hop address to its own address when advertising routes to the IBGP peer groups, regardless of whether the **peer next-hop-local** command is used.

- The next-hop iteration based on the routing policy

Do as follows on the BGP router:

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
bgp as-number
```

The BGP view is displayed.

3. Run:

```
ipv6-family [ unicast ]
```

The BGP-IPv6 unicast address family view is displayed.

4. Run:

```
nexthop recursive-lookup route-policy route-policy-name
```

The next-hop iteration based on the specified routing policy is enabled.

By default, the next-hop iteration based on the specified routing policy is disabled.

The next-hop iteration based on the specified routing policy can control the iterated route according to certain conditions. The route that fails to pass the policy is ignored.

---End

9.4.7 Configuring the AS-Path Attribute

The AS_Path attribute is used to avoid routing loops and control route selection.

Procedure

- Configuring the AS_Path Attribute in the IPv6 Address Family View

Do as follows on the BGP4+ router:

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
bgp as-number
```

The BGP view is displayed.

3. Run:

```
ipv6-family [ unicast ]
```

The BGP IPv6 unicast address family view is displayed.

4. Run the following commands to configure the AS-Path attribute as required:

- Run:

```
peer { ipv6-address | group-name } allow-as-loop [ number ]
```

The local AS number can be used repeatedly.

- Run:

```
bestroute as-path-ignore
```

The AS-Path attribute is not configured as one of the route selection rules.

- Run:

```
peer { ipv6-address | group-name } public-as-only
```

The AS-Path attribute is configured to carry only the public AS number.

The commands in Step 4 can be used regardless of the order.

- Configuring the Fake AS Number

Do as follows on the BGP4+ router:

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
bgp as-number
```

The BGP view is displayed.

3. Run:

```
peer { ipv6-address | group-name } fake-as fake-as-number
```

The fake AS number is set.

You can hide the actual AS number of the local router by using this command. EBGP peers in other ASs can only see this fake AS number. That is, peers in other ASs need to specify the number of the AS where the local peer resides as this fake AS number.

 **NOTE**

This command is applicable only to EBGP peers.

---End

9.4.8 Configuring the BGP4+ Community Attribute

The community attribute is used to simplify the management of routing policies. The management scope of the community attribute is far larger than that of the peer group. The community attribute can control the routing policies of multiple BGP4+ routers.

Procedure

- Configuring the routers to Advertise the Community Attribute to the Peers

Do as follows on the BGP4+ router:

1. Run:

```
system-view
```

The system view is displayed.

2. Run:
`bgp as-number`

The BGP view is displayed.

3. Run:
`ipv6-family [unicast]`

The BGP IPv6 unicast address family view is displayed.

4. Run the following commands to advertise community attributes to the peer group:

- Run:

```
peer { ipv6-address | group-name } advertise-community
```

routers are configured to advertise the standard community attribute to a peer group.

- Run:

```
peer { ipv6-address | group-name } advertise-ext-community
```

routers are configured to advertise the extended community attribute to a peer group.

- Applying the Routing Policies to the Advertised Routing Information

Do as follows on the BGP4+ router:

1. Run:
`system-view`

The system view is displayed.

2. Run:
`bgp as-number`

The BGP view is displayed.

3. Run:
`ipv6-family [unicast]`

The BGP IPv6 unicast address family view is displayed.

4. Run:
`peer { ipv6-address | group-name } route-policy route-policy-name export`

The outbound routing policies are configured.

NOTE

- When configuring a BGP4+ community, you should define the specific community attribute by using the routing policies. Then, apply these routing policies to the advertisement of routing information.
- For the configuration of routing policies, refer to [Routing Policy Configuration](#). For the configuration of community attributes, refer to [8 BGP Configuration](#).

---End

9.4.9 Checking the Configuration

After BGP4+ route attributes are configured, you can check information about route attributes.

Prerequisites

The configurations for BGP4+ route attributes are complete.

Procedure

- Run the **display bgp ipv6 paths** [*as-regular-expression*] command to check the AS-Path information.
- Run the **display bgp ipv6 routing-table different-origin-as** command to check the route with the different source AS.
- Run the **display bgp ipv6 routing-table regular-expression** *as-regular-expression* command to check the routing information matching the regular expression of the AS.
- Run the **display bgp ipv6 routing-table community** [*aa:nn &<1-29>*] [**internet** | **no-advertise** | **no-export** | **no-export-subconfed**] * [**whole-match**] command to check routing information about the specified BGP4+ community.
- Run the **display bgp ipv6 routing-table community-filter** { { *community-filter-name* | *basic-community-filter-number* } [**whole-match**] | *advanced-community-filter-number* } command to check information about the routes matching the specified BGP4+ community attribute filter.

----End

9.5 Controlling the Advertising and Receiving of BGP4+ Routing Information

BGP4+ can perform routing policies on or filter only the routes to be advertised to a certain peer.

9.5.1 Establishing the Configuration Task

Before controlling the advertisement of BGP4+ routes, familiarize yourself with the applicable environment, complete the pre-configuration tasks, and obtain the required data. This can help you complete the configuration task quickly and accurately.

Applicable Environment

This section describes the following:

- Controlling the advertising and receiving of BGP4+ routing information, which includes the filtering of routing information and the application of the routing policies.
- Soft resetting the BGP4+ connections

In the AR150/200, BGP4+ supports the route-refresh capability. When the policies are changed, the system can refresh the BGP4+ routing table automatically without interrupting the BGP4+ connections.

If there are routers that do not support route-refresh in the network, you can run the **peer keep-all-routes** command to save all route refreshment locally. Then, you can run the **refresh bgp** command to soft reset the BGP4+ connections manually.

Pre-configuration Tasks

Before controlling the advertising and receiving of BGP4+ routing information, complete the following tasks:

- **Configuring Basic BGP4+ Functions**

Data Preparation

To control the advertising and receiving of BGP4+ routing information, you need the following data.

No.	Data
1	Name and process ID of the external route to be imported
2	Name of the filtering list used in the routing policies
3	Various parameters of route dampening, including half-life of a reachable route, half-life of an unreachable route, threshold for freeing suppressed routes, threshold for suppressing routes, and upper limit of the penalty

9.5.2 Configuring BGP4+ to Advertise Local IPv6 Routes

The local routes to be advertised must be in the local IP routing table. You can use routing policies to control the routes to be advertised.

Context

Do as follows on the BGP4+ router:

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
bgp as-number
```

The BGP view is displayed.

Step 3 Run:

```
ipv6-family [ unicast ]
```

The BGP IPv6 unicast address family view is displayed.

Step 4 Run:

```
network ipv6-address prefix-length [ route-policy route-policy-name ]
```

The exactly-matched local IPv6 routes are advertised.

You can use the **network** command to statically inject the IPv6 routes to the BGP4+ routing table.

To be specific, the command can be used to advertise the routes only with the exactly-matched address prefix and mask. If the mask is not designated, the routes are exactly matched based on the natural network segment.

The local routes to be advertised should be in the local IPv6 routing table. You can use routing policies to control the routes to be advertised more flexibly.

---End

9.5.3 Configuring BGP4+ Route Aggregation

By configuring route aggregation, you can reduce the size of the routing table of a peer. BGP4+ supports automatic aggregation and manual aggregation.

Context

Do as follows on the router enabled with BGP4+:

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
bgp as-number
```

The BGP view is displayed.

Step 3 Run:

```
ipv6-family [ unicast ]
```

The IPv6 unicast address family view is displayed.

Step 4 Run:

```
aggregate ipv6-address prefix-length [ as-set | attribute-policy route-policy-name1 | detail-suppressed | origin-policy route-policy-name2 | suppress-policy route-policy-name3 ] *
```

Manual aggregation of routes is configured.

Manual aggregation is valid for the routing entries in the local BGP4+ routing table. For example, if 9:3::1/64 does not exist in the BGP routing table, BGP4+ does not advertise the aggregated route even after the **aggregate 9:3::1 64** command is run to aggregate this route.

When configuring manual aggregation of routes, you can apply various routing policies and set the route attributes.

---End

9.5.4 Configuring BGP4+ to Import and Filter External Routes

After BGP4+ filters the imported routes, only the eligible routes are added to the local BGP4+ routing table and advertised to BGP4+ peers.

Context

Do as follows on the BGP4+ router:

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
bgp as-number
```

The BGP view is displayed.

Step 3 Run:

```
ipv6-family [ unicast ]
```

The BGP IPv6 unicast address family view is displayed.

Step 4 Run:

```
default-route imported
```

BGP4+ is configured to import the default routes.

If the **default-route imported** command is not used, you cannot import the default routes from other protocols by using the **import-route** command.

Step 5 Run:

```
import-route protocol [ process-id ] [ med med | route-policy route-policy-name ] *
```

BGP4+ is configured to import routes of other protocols.

 **NOTE**

Specify the process ID when the routes of a dynamic routing protocol are imported.

Step 6 Run:

```
filter-policy ipv6-prefix-name export [ protocol [ process-id ] ]
```

Imported routes are filtered.

After BGP4+ filters the imported routes, only the eligible routes are added to the BGP4+ local routing table and advertised to BGP4+ peers. If *protocol [process-id]* is specified, the routes of the specific routing protocol are filtered. If *protocol [process-id]* is not specified, all the local BGP routes to be advertised are filtered, including the imported routes and the local routes advertised through the **network** command.

----End

9.5.5 Configuring s to Advertise Default Routes to Peers

A router sends a default route with the local address being the next hop to the specified peer, regardless of whether there are default routes in the local routing table.

Context

Do as follows on the BGP4+ router:

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
bgp as-number
```

The BGP view is displayed.

Step 3 Run:

```
ipv6-family unicast
```

The BGP IPv6 unicast address family view is displayed.

Step 4 Run:

```
peer { ipv6-address | group-name } default-route-advertise [ route-policy route-policy-name ]
```

Default routes are advertised to peers (or a peer group).

 **NOTE**

After the command **peer default-route-advertise** is run, the router sends a default route with the local address as the next hop to the specified peer, regardless of whether there are default routes in the routing table.

----End

9.5.6 Configuring the Policy for Advertising BGP4+ Routing Information

After the policy for advertising routes is configured, only the routes that match the policy can be added to the local BGP4+ routing table and advertised to BGP4+ peers.

Context

Do as follows on the BGP4+ router:

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
bgp as-number
```

The BGP view is displayed.

Step 3 Run:

```
ipv6-family [ unicast ]
```

The BGP IPv6 unicast address family view is displayed.

Step 4 Run the following command to configure the outbound routing policy based on the following different filters:

- Based on the routing policy

Run:

```
peer { ipv6-address | group-name } route-policy route-policy-name export
```

- Based on the AS_Path list

Run:

```
peer { ipv6-address | group-name } as-path-filter { as-path-filter-number | as-path-filter-name } export
```

- Based on the prefix list

Run:

```
peer { ipv6-address | group-name } ipv6-prefix ip-prefix-name export
```

The commands in Step 4 can be run regardless of the order.

The outbound routing updates policies used by the members of a peer group can be different from that used by the group. That is, members of each peer group can select their policies when advertising routes externally.

----End

9.5.7 Configuring the Policy for Receiving BGP4+ Routing Information

Only the routes that match the policy for receiving routes can be received by BGP4+ peers and added to the routing table.

Context

Do as follows on the BGP4+ router:

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
bgp as-number
```

The BGP view is displayed.

Step 3 Run:

```
ipv6-family [ unicast ]
```

The BGP IPv6 unicast address family view is displayed.

Step 4 Run:

- **filter-policy { acl6-number | acl6-name acl6-name | ipv6-prefix ipv6-prefix-name } import**
The imported global routes are filtered.

- **peer { ipv6-address | group-name } route-policy route-policy-name import**
BGP is configured to filter the routes imported from the specified peers.

- **peer { ipv6-address | group-name } as-path-filter { as-path-filter-number | as-path-filter-name } import**

BGP is configured to filter the routes based on the AS path list.

- **peer { ipv6-address | group-name } ipv6-prefix ipv6-prefix-name import**

BGP is configured to filter the routes based on the prefix list.

The commands in Steps 4 can be run regardless of the order.

The routes imported by BGP can be filtered, and only those routes that meet certain conditions are received by BGP and added to the routing table.

The inbound routing policies used by the members in a peer group can be different from that used by the group. That is, each peer can select its policy when importing routes.

---End

9.5.8 Configuring BGP4+ Soft Resetting

When routing policies are changed, the system can refresh the BGP4+ routing table dynamically without interrupting BGP4+ connections.

Procedure

- Enabling the Route-refresh Capability

Do as follows on the BGP4+ router:

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
bgp as-number
```

The BGP view is displayed.

3. Run:

```
peer { ipv6-address | group-name } capability-advertise { route-refresh | 4-byte-as }
```

The route-refresh capability is enabled.

By default, the route-refresh capability is enabled.

If the route-refresh capability is enabled on all the BGP4+ routers, the local router advertises the route-refresh messages to its peer if the BGP4+ route policies change. The peer receiving this message sends its routing information to the local router again. In this way, the BGP4+ routing table is updated dynamically and the new policies are applied without interrupting the BGP4+ connections.

- Keeping All Route Updates of Peers

Do as follows on the BGP4+ router:

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
bgp as-number
```

The BGP view is displayed.

3. Run:

```
ipv6-family [ unicast ]
```

The BGP IPv6 unicast address family view is displayed.

4. Run:

```
peer { ipv6-address | group-name } keep-all-routes
```

All route updates of the peers are kept.

After this command is run, all the route updates of the specified peer are kept regardless of whether the filtering policies are used. When the BGP connections are soft reset, this information can be used to generate the BGP4+ routes.

- Soft Resetting a BGP4+ Connection Manually

Do as follows on the BGP4+ router:

1. Run:

```
refresh bgp ipv6 { all | ipv6-address | group group-name | external |  
internal } { export | import }
```

A BGP4+ connection is soft reset.

A BGP4+ connection must be soft reset in the user view.

----End

9.5.9 Checking the Configuration

After the advertising and receiving of BGP4+ routes are controlled, you can check the advertised routes that match the specified filter.

Prerequisites

The configurations for controlling the advertising and receiving of BGP4+ routing information are complete.

Procedure

- Run the **display bgp ipv6 network** command to check the routes advertised through the **network** command.
- Run the **display bgp ipv6 routing-table as-path-filter** { *as-path-filter-number* | *as-path-filter-name* } command to check the routes matching the specified AS-Path filter.
- Run the **display bgp ipv6 routing-table community-filter** { { *community-filter-name* | *basic-community-filter-number* } [**whole-match**] | *advanced-community-filter-number* } command to check the routes matching the specified BGP4+ community filter.
- Run the **display bgp ipv6 routing-table peer** *ipv6-address* { **advertised-routes** | **received-routes** } [**statistics**] command to check the routing information advertised or received by the BGP4+ peers.

----End

9.6 Configuring Parameters of a Connection Between BGP4+ Peers

By setting parameters of a connection between BGP4+ peers, you can adjust and optimize the BGP4+ network performance.

9.6.1 Establishing the Configuration Task

Before configuring parameters of a connection between BGP4+ peers, familiarize yourself with the applicable environment, complete the pre-configuration tasks, and obtain the required data. This can help you complete the configuration task quickly and accurately.

Applicable Environment

After a BGP4+ connection is set up between peers, the peers periodically send Keepalive messages to each other. This prevents the routers from considering that the BGP4+ connection is closed. If a router does not receive any Keepalive message or any type of packets from the peer within the specified Hold time, the BGP4+ connection is considered as closed.

When a router sets up a BGP4+ connection with its peer, the router and the peer need negotiation with each other. The Hold time after negotiation is the shorter one between the Hold time of the router and that of its peer. If the negotiation result is 0, no Keepalive message is transmitted and whether the Hold timer expires is not detected.

If the value of the timer changes, the BGP4+ connection is interrupted for a short time as the router and its peer need negotiate again.

A ConnectRetry timer is used to set the interval between BGP4+ attempts to initiate TCP connections. After BGP4+ initiates a TCP connection, the ConnectRetry timer is stopped if the TCP connection is established successfully. If the first attempt to establish a TCP connection fails, BGP4+ tries again to establish the TCP connection after the ConnectRetry timer expires.

You can speed up or slow down the establishment of BGP4+ peer relationships by changing the BGP4+ ConnectRetry interval. For example, if the ConnectRetry interval is reduced, BGP4+ will wait less time to retry establishing a TCP connection when an earlier attempt fails. This speeds up the establishment of the TCP connection. If a BGP4+ peer flaps constantly, the ConnectRetry interval can be increased to suppress route flapping caused by BGP4+ peer flapping. This speeds up route convergence.

Pre-configuration Tasks

Before configuring the parameters of a connection between BGP4+ peers, complete the following tasks:

- [Configuring Basic BGP4+ Functions](#)

Data Preparation

To configure the parameters of a connection between BGP4+ peers, you need the following data.

No.	Data
1	Values of the BGP4+ timers
2	Interval for sending the update packets
3	BGP4+ ConnectRetry interval

9.6.2 Configuring BGP4+ Timers

Configuring timers properly can improve network performance. Changing the values of BGP4+ timers will interrupt the peer relationship.

Context



CAUTION

As the change of the timer (with the **peer timer** command) tears down the BGP peer relationship between routers. Exercise caution when running this command.

Do as follows on the BGP4+ router:

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
bgp as-number
```

The BGP view is displayed.

Step 3 Run:

```
peer { ipv6-address | group-name } timer keepalive keepalive-time hold hold-time
```

The interval for sending Keepalive messages and the Hold time of a peer (or peer group) are set.

In actual applications, the value of *hold-time* is at least three times that of *keepalive-time*.

By default, the Keepalive time is 60s and the Hold time is 180s.

NOTE

Setting the hold interval of a BGP peer to be longer than 20s is recommended. If the hold interval of a BGP peer is shorter than 20s, the session may be closed.

Note the following when you set the values of *keepalive-time* and *hold-time*:

- When the values of *keepalive-time* and *hold-time* are 0 at the same time, the BGP timer becomes invalid. That is, BGP does not detect link faults according to the timer.
- The value of *hold-time* is far greater than that of *keepalive-time*, such as, **timer keepalive 1 hold 65535**. If the Hold time is too long, the link fault cannot be detected on time.

----End

9.6.3 Configuring the Interval for Sending Update Packets

When a route changes, a router sends an Update packet to notify its peer. If a route changes frequently, to prevent the router from sending Update packets for every change, you can set the interval for sending Update packets for changes of this route.

Context

Do as follows on the BGP4+ router:

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
bgp as-number
```

The BGP view is displayed.

Step 3 Run:

```
ipv6-family [ unicast ]
```

The BGP IPv6 unicast address family view is displayed.

Step 4 Run:

```
peer ipv6-address route-update-interval interval
```

The interval for sending update packets is set.

By default, the update interval is 15 seconds for the IBGP peers and the update interval is 30 seconds for the EBGP peers.

----End

9.6.4 Setting the BGP4+ ConnectRetry Interval

You can speed up or slow down the establishment of BGP4+ peer relationships to adapt the network changes by changing the BGP4+ ConnectRetry interval.

Context

When BGP4+ initiates a TCP connection, the ConnectRetry timer is stopped if the TCP connection is established successfully. If the first attempt to establish a TCP connection fails, BGP4+ tries again to establish the TCP connection after the ConnectRetry timer expires. The ConnectRetry interval can be adjusted as needed.

- The ConnectRetry interval can be reduced in order to lessen the time BGP4+ waits to retry establishing a TCP connection after the first attempt fails.
- To suppress route flapping caused by constant peer flapping, the ConnectRetry interval can be increased to speed up route convergence.

Do as follows on the BGP4+ router:

Procedure

- Set a ConnectRetry interval globally.

Do as follows on the BGP4+ router:

1. Run:

```
system-view
```

- The system view is displayed.
2. Run:
`bgp as-number`
The BGP view is displayed.
 3. Run:
`timer connect-retry connect-retry-time`
A BGP4+ ConnectRetry interval is set globally.
By default, the ConnectRetry interval is 32s.
- Set a ConnectRetry interval on a peer or peer group.
Do as follows on the BGP4+ router:
 1. Run:
`system-view`
The system view is displayed.
 2. Run:
`bgp as-number`
The BGP view is displayed.
 3. Run:
`peer { group-name | ipv6-address } timer connect-retry connect-retry-time`
A ConnectRetry interval is set on a peer or peer group.
By default, the ConnectRetry interval is 32s.
The ConnectRetry interval configured on a peer or peer group takes precedence over a global ConnectRetry interval.
- End

9.6.5 Checking the Configuration

After parameters of a connection between BGP4+ peers are configured, you can check BGP4+ peers and peer groups.

Prerequisites

The configurations for parameters of a connection between BGP4+ peers are complete.

Procedure

- Run the `display bgp ipv6 peer ipv6-address { log-info | verbose }` command to check information about the BGP4+ peers.

----End

9.7 Configuring BGP4+ Tracking

On a network where BFD is unsuitable to deploy, you can configure BGP4+ tracking to implement the fast convergence of IBGP routes.

9.7.1 Establishing the Configuration Task

Before configuring BGP4+ tracking, familiarize yourself with the applicable environment, complete the pre-configuration tasks, and obtain the required data. This can help you complete the configuration task quickly and accurately.

Applicable Environment

Since BFD is difficult to deploy and is of poor scalability, in a network where BFD is unsuitable to be deployed, you can configure BGP4+ tracking as a substitution for BFD to implement the fast convergence of BGP4+ routes.

BGP4+ tracking is easy to deploy because it needs to be configured only on the local device, without the need of configuring it on the peer device. However, BGP4+ route convergence in a network configured with BGP4+ tracking is slower than that in a network enabled with BFD; therefore, BGP4+ tracking cannot meet the requirement of voice services that demand high convergence speed.

Pre-configuration Tasks

Before configuring BGP4+ tracking, complete the following tasks:

- [Configuring basic BGP4+ functions](#)

Data Preparation

To configure BGP4+ tracking, you need the following data.

No.	Data
1	(Optional) Delay for tearing down a connection

9.7.2 Enabling BGP4+ Tracking

Easy to deploy, BGP4+ tracking can speed up network convergence and adjust the interval between a peer's being discovered unreachable and the connection's being torn down.

Context

Do as follows on the router enabled with BGP4+:

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
bgp as-number
```

The BGP view is displayed.

Step 3 Run:

```
peer { group-name | ipv6-address } tracking [ delay delay-time ]
```

BGP4+ tracking is enabled for the specified peer.

By default, BGP4+ tracking is disabled.

A proper value of *delay-time* can ensure network stability when a peer is detected unreachable.

- If *delay-time* is set to 0, BGP immediately tears down the connection between the local device and its peer after the peer is detected unreachable.
- If IGP route flapping occurs and *delay-time* for an IBGP peer is set to 0, the peer relationship between the local device and the peer alternates between Up and Down. Therefore, *delay-time* for an IBGP peer should be set to a value greater than the actual IGP route convergence time.
- When BGP neighbors successfully perform the GR negotiation, the active/standby switchover occurs on the BGP neighbors, to prevent the failure of GR, *delay-time* should be set to a value greater than GR period. If *delay-time* is set to be smaller than the GR period, the connection between the local device and the BGP peer will be torn down, which leads to the failure of GR.

---End

9.7.3 Checking the Configuration

After BGP4+ tracking is configured, you can check the configuration of BGP4+ tracking by viewing detailed information about the BGP peer or peer group.

Prerequisite

All BGP4+ tracking configurations are complete.

Checking the Configuration

Run the following commands to check the previous configuration.

- Run the **display bgp ipv6 peer** [[*ipv6-address*] **verbose**] command to check information about the BGP4+ peer.
- Run the **display bgp ipv6 group** [*group-name*] command to check information about the BGP4+ peer group.

9.8 Configuring BGP4+ Route Dampening

By configuring BGP4+ route dampening, you can suppress unstable BGP4+ routes.

9.8.1 Establishing the Configuration Task

Before configuring BGP4+ route dampening, familiarize yourself with the applicable environment, complete the pre-configuration tasks, and obtain the required data. This can help you complete the configuration task quickly and accurately.

Applicable Environment

BGP4+ dampening can suppress unstable routes. BGP4+ neither adds the unstable routes to the routing table nor advertises them to other BGP peers.

Pre-configuration Tasks

Before configuring BGP4+ route dampening, complete the following task:

- [Configuring Basic BGP4+ Functions](#)

Data Preparation

To configure BGP4+ route dampening, you need the following data.

No.	Data
1	Various parameters of dampening, including half-life of a reachable route, half-life of an unreachable route, threshold for freeing the suppressed routes, threshold for suppressing routes, and upper limit of the penalty

9.8.2 Enabling BGP4+ Route Dampening

BGP4+ route dampening can improve network stability. You can flexibly use routing policies for route dampening.

Context

Do as follows on the BGP4+ router:

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
bgp as-number
```

The BGP view is displayed.

Step 3 Run:

```
ipv6-family [ unicast ]
```

The BGP IPv6 unicast address family view is displayed.

Step 4 Run:

```
dampening [ half-life-reach reuse suppress ceiling | route-policy route-policy-name ] *
```

The parameters are configured for BGP4+ route dampening.

----End

9.8.3 Checking the Configuration

After BGP4+ route dampening is configured, you can check BGP4+ suppressed routes, parameters of BGP4+ route dampening, and flapped routes.

Prerequisites

The configurations for BGP4+ route dampening are complete.

Procedure

- Run the **display bgp ipv6 routing-table dampened** command to check BGP4+ dampened routes.
- Run the **display bgp ipv6 routing-table dampening parameter** command to check the configuration parameters of BGP4+ dampening.
- Run the **display bgp ipv6 routing-table flap-info [regular-expression *as-regular-expression* | as-path-filter { *as-path-filter-number* | *as-path-filter-name* } | network-address [*prefix-length* [**longer-match**]]]** command to check the statistics of BGP4+ route flapping.

----End

9.9 Configuring BGP4+ Load Balancing

Configuring BGP4+ load balancing better utilizes network resources and reduces network congestion.

Applicable Environment

On large networks, there may be multiple valid routes to the same destination. BGP, however, advertises only the optimal route to its peers. This may result in unbalanced traffic on different routes.

Either of the following methods can be used to address the problem of unbalanced traffic:

- Use BGP routing policies to allow traffic to be balanced. For example, use a routing policy to modify the Local_Pref, AS_Path, Origin, and Multi Exit Discriminator (MED) attributes of BGP routes to direct traffic to different forwarding paths for load balancing. For details on how to modify attributes of BGP routes, see [Configuring BGP4+ Route Attributes](#).
- Use multiple paths for load balancing. In this method, multiple equal-cost routes need to be configured for traffic load balancing.

NOTE

Equal-cost BGP routes can be generated for traffic load balancing only when the first 8 route attributes described in "Route Selection Policies for Load Balancing" in [BGP Features Supported by the AR150/200](#) are the same, and the AS-Path attributes are also the same.

Pre-configuration Tasks

Before configuring BGP4+ load balancing, complete the following task:

- [Configuring Basic BGP4+ Functions](#)

Data Preparation

To configure BGP4+ load balancing, you need the following data.

No.	Data
1	Number of BGP4+ routes to be used for load balancing

Procedure

- Set the number of BGP4+ routes to be used for load balancing.

Perform the following steps on a BGP4+ router:

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
bgp as-number
```

The BGP view is displayed.

3. Run:

```
ipv6-family [ unicast ]
```

The IPv6 unicast address family view is displayed.

4. Run:

```
maximum load-balancing [ ebgp | ibgp ] number
```

The number of BGP4+ routes to be used for load balancing is set.

By default, the number of BGP4+ routes to be used for load balancing is 1, meaning that load balancing is not implemented.

- **ebgp** indicates that load balancing is implemented only among EBGp routes.
- **ibgp** indicates that load balancing is implemented only among IBGP routes.
- If neither **ebgp** nor **ibgp** is specified, both EBGp and IBGP routes participate in load balancing, and the number of EBGp routes to be used for load balancing is the same as the number of IBGP routes to be used for load balancing.

NOTE

The **maximum load-balancing number** command cannot be configured together with the **maximum load-balancing ebgp number** or **maximum load-balancing ibgp number** command.

When routes with the same destination addresses carry out load balancing on the public network, the system determines the type of optimal routes first. If the optimal routes are IBGP routes, only IBGP routes carry out load balancing. If the optimal routes are EBGp routes, only EBGp routes carry out load balancing. This means that load balancing cannot be implemented among IBGP and EBGp routes with the same destination address.

5. (Optional) Run:

```
load-balancing as-path-ignore
```

The router is configured not to compare the AS-Path attributes of the routes to be used for load balancing.

By default, the router compares the AS-Path attributes of the routes to be used for load balancing.

 **NOTE**

- If there are multiple routes to the same destination but these routes pass through different ASs, load balancing cannot be implemented among these routes by default. To implement load balancing among these routes, run the **load-balancing as-path-ignore** command. After the **load-balancing as-path-ignore** command is run, the device no longer compares the AS-Path attributes of the routes to be used for load balancing. Therefore, exercise caution when using this command.
- The **load-balancing as-path-ignore** and **bestroute as-path-ignore** commands are mutually exclusive.

----End

Checking the Configuration

After the BGP4+ load balancing configurations are complete, you can run the following commands to check the configurations.

- Run the **display bgp ipv6 routing-table** [*ipv6-address prefix-length*] command to check routing information in an IPv6 BGP routing table.
- Run the **display ipv6 routing-table** [*verbose*] command to view the IPv6 routing table.

9.10 Configuring a BGP4+ Peer Group

By configuring a BGP4+ peer group, you can simplify the management of routing policies, and thus improve the efficiency of route advertisement.

9.10.1 Establishing the Configuration Task

Before configuring a BGP4+ peer group, familiarize yourself with the applicable environment, complete the pre-configuration tasks, and obtain the required data. This can help you complete the configuration task quickly and accurately.

Applicable Environment

A great number of peers exist in a large-scale BGP4+ network, which is not convenient for configuration and maintenance. In this case, you can configure peer groups to simplify the management and improve the efficiency of route advertisement. According to the AS where the peers reside, you can classify peer groups into IBGP peer groups and EBGP peer groups. You can classify EBGP peer groups into pure EBGP peer groups and mixed EBGP peer groups. This classification is performed according to the position of the peers in the same external AS.

Pre-configuration Tasks

Before configuring a BGP4+ peer group, complete the following task:

- **Configuring Basic BGP4+ Functions**

Data Preparation

To configure a BGP4+ peer group, you need the following data.

No.	Data
1	Type, name of the peer group, and the member peers

9.10.2 Creating an IBGP Peer Group

When BGP4+ has multiple IBGP peers, you can create an IBGP peer group to simplify the management of routing policies. When creating an IBGP peer group, you do not need to specify the AS number.

Context

Do as follows on the BGP4+ router:

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
bgp as-number
```

The BGP view is displayed.

Step 3 Run:

```
group group-name [ internal ]
```

A peer group is created.

Step 4 Run:

```
ipv6-family [ unicast ]
```

The BGP IPv6 unicast address family view is displayed.

Step 5 Run:

```
peer group-name enable
```

The peer group is enabled.

Step 6 Run:

```
peer ipv6-address group group-name
```

The IPv6 peers are added to the peer group.

 **NOTE**

After an IBGP peer is added to a peer group, the system automatically creates the IPv6 peer in the BGP view. Besides, the system enables this IBGP peer in the IPv6 address family view.

----End

9.10.3 Creating a Pure EBGP Peer Group

When BGP4+ has multiple EBGP peers that belong to the same AS, you can create an EBGP peer group to simplify the management of routing policies. All the peers in a pure EBGP peer group must have the same AS number.

Context

Do as follows on the BGP4+ router:

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
bgp as-number
```

The BGP view is displayed.

Step 3 Run:

```
group group-name external
```

A pure EBGP peer group is configured.

Step 4 Run:

```
peer group-name as-number as-number
```

The AS number of the peer group is set.

Step 5 Run:

```
ipv6-family [ unicast ]
```

The BGP IPv6 unicast address family view is displayed.

Step 6 Run:

```
peer group-name enable
```

The peer group is enabled.

Step 7 Run:

```
peer ipv6-address group group-name
```

The IPv6 peer is added to the peer group.

After an EBGP peer is added to the peer group, the system automatically creates the EBGP peer in the BGP view. Besides, the system enables this EBGP peer in the IPv6 address family view.

When creating a pure EBGP peer group, you need to specify the AS number of the peer group.

If there are peers in the peer group, you cannot specify the AS number for this peer group.

----End

9.10.4 Creating a Mixed EBGP Peer Group

When BGP4+ has multiple EBGP peers that belong to different ASs, you can create a mixed EBGP peer group to simplify the management of routing policies. When creating a mixed EBGP peer group, you need to specify the AS number for each peer.

Context

Do as follows on the BGP4+ router:

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
bgp as-number
```

The BGP view is displayed.

Step 3 Run:

```
group group-name external
```

A mixed EBGP peer group is created.

Step 4 Run:

```
peer ipv6-address as-number as-number
```

The AS number of the IPv6 peer is set.

Step 5 Run:

```
ipv6-family [ unicast ]
```

The BGP IPv6 unicast address family view is displayed.

Step 6 Run:

```
peer group-name enable
```

The peer group is enabled.

Step 7 Run:

```
peer ipv6-address group group-name
```

The IPv6 peers created are added to this peer group.

After an EBGP peer is added to the peer group, the system automatically enables each EBGP peer in the IPv6 address family view.

When creating a mixed EBGP peer group, you need to create peers separately, and you can configure different AS numbers for them, but cannot configure the AS number for the peer group.

----End

9.10.5 Checking the Configuration

After a BGP4+ peer group is configured, you can check detailed information about the BGP4+ peer and information about the BGP4+ peer group.

Prerequisites

The configurations for a BGP4+ peer group are complete.

Procedure

- Run the **display bgp ipv6 group** [*group-name*] command to check information about the IPv6 peer group.

---End

9.11 Configuring a BGP4+ Route Reflector

By configuring a BGP4+ route reflector, you can solve the problem of establishing fully meshed connections between multiple IBGP peers.

9.11.1 Establishing the Configuration Task

Before configuring a BGP4+ route reflector, familiarize yourself with the applicable environment, complete the pre-configuration tasks, and obtain the required data. This can help you complete the configuration task quickly and accurately.

Applicable Environment

To ensure the connectivity between IBGP peers inside an AS, you need to establish full-meshed IBGP peers. When there are many IBGP peers, establishing a full-meshed network costs a lot. The route reflector or the confederation can be used to solve this problem.

Pre-configuration Tasks

Before configuring a BGP4+ route reflector, complete the following task:

- [9.3 Configuring Basic BGP4+ Functions](#)

Data Preparation

To configure a BGP4+ route reflector, you need the following data.

No.	Data
1	Roles of each router (reflector, client, and non-client)

9.11.2 Configuring a Route Reflector and Specifying Clients

A route reflector and clients need to be configured in a specified address family.

Context

Do as follows on the BGP4+ router:

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
bgp as-number
```

The BGP view is displayed.

Step 3 Run:

```
ipv6-family [ unicast ]
```

The BGP IPv6 unicast address family view is displayed.

Step 4 Run:

```
peer { ipv6-address | group-name } reflect-client
```

The route reflector and its clients are configured.

The router on which this command is run serves as the route reflector. In addition, this command specifies the peers that serve as its clients.

----End

9.11.3 (Optional) Disabling a Route Reflection Between Clients

If the clients of a route reflector are fully meshed, you can disable route reflection between clients to reduce the cost.

Context

Do as follows on the BGP4+ router:

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
bgp as-number
```

The BGP view is displayed.

Step 3 Run:

```
ipv6-family [ unicast ]
```

The BGP IPv6 unicast address family view is displayed.

Step 4 Run:

```
undo reflect between-clients
```

Route reflection between clients is disabled.

If the clients of the route reflector are full-meshed, you can use the **undo reflect between-clients** command to disable the route reflection between the clients. This reduces cost.

By default, the route reflection between clients is enabled.

This command is used only on the reflector.

----End

9.11.4 (Optional) Configuring the Cluster ID for a Route Reflector

When there are multiple route reflectors in a cluster, you need to configure the same cluster ID for all the route reflectors in this cluster to avoid routing loops.

Context

Do as follows on the BGP4+ router:

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
bgp as-number
```

The BGP view is displayed.

Step 3 Run:

```
ipv6-family [ unicast ]
```

The BGP IPv6 unicast address family view is displayed.

Step 4 Run:

```
reflector cluster-id cluster-id
```

The cluster ID of the route reflector is set.

 **TIP**

When there are multiple route reflectors in a cluster, you can use the command to configure all the route reflectors in this cluster with the same cluster ID. This avoids routing loops.

----End

9.11.5 Checking the Configuration

After a BGP4+ route reflector is configured, you can check BGP4+ route information and peer group information.

Prerequisites

The configurations for a BGP4+ route reflector are complete.

Procedure

- Run the **display bgp ipv6 peer [verbose]** command to check information about BGP4+ peers.

- Run the **display bgp ipv6 peer** *ipv6-address* { **log-info** | **verbose** } command to check information about BGP4+ peers.

---End

9.12 Configuring a BGP4+ Confederation

On a large-scale BGP4+ network, configuring a BGP4+ confederation can simplify the management of routing policies and improve the efficiency of route advertisement.

9.12.1 Establishing the Configuration Task

Before configuring a BGP4+ confederation, familiarize yourself with the applicable environment, complete the pre-configuration tasks, and obtain the required data. This can help you complete the configuration task quickly and accurately.

Applicable Environment

The confederation is a method of handling the abrupt increase of IBGP connections in an AS. The confederation divides an AS into multiple sub-ASs. In each sub-AS, IBGP peers can be full-meshed or be configured with a route reflector. EBGP connections are set up between sub-ASs.

Pre-configuration Tasks

Before configuring a BGP4+ confederation, complete the following task:

- [Configuring Basic BGP4+ Functions](#)

Data Preparation

To configure a BGP4+ confederation, you need the following data.

No.	Data
1	Confederation ID and the sub-AS number

9.12.2 Configuring a BGP4+ Confederation Attribute

BGP4+ confederations deal with increasing IBGP connections in an AS.

Context

Do as follows on the BGP4+ router:

Procedure

- Configuring a BGP4+ Confederation
 1. Run:
`system-view`

The system view is displayed.

2. Run:
`bgp as-number`

The BGP view is displayed.

3. Run:
`confederation id as-number`

The confederation ID is set.

4. Run:
`confederation peer-as as-number <1-32>`

The sub-AS number of other EBGP peers connected with the local AS is set.

A confederation includes up to 32 sub-ASs. *as-number* is valid for the confederation that it belongs to.

You must run the **confederation id** and **confederation peer-as** commands for all the EBGP peers that belong to a confederation, and specify the same confederation ID for them.

 **NOTE**

The old speaker with 2-byte AS numbers and the new speaker with 4-byte AS numbers cannot exist in the same confederation. Otherwise, routing loops may occur because AS4_Path does not support confederations.

- **Configuring the Compatibility of a Confederation**

1. Run:
`system-view`

The system view is displayed.

2. Run:
`bgp as-number`

The BGP view is displayed.

3. Run:
`confederation nonstandard`

The compatibility of the confederation is configured.

When the confederation of other routers does not conform to the RFC, you can use this command to make standard devices be compatible with nonstandard devices.

----End

9.12.3 Checking the Configuration

After a BGP4+ confederation is configured, you can check BGP4+ route information and detailed peer information.

Prerequisites

The configurations for a BGP4+ confederation are complete.

Procedure

- Run the **display bgp ipv6 peer [verbose]** command to check detailed information about BGP4+ peers.

----End

9.13 Configuring BGP4+ Security

To improve BGP4+ security, you can perform TCP connection authentication.

9.13.1 Establishing the Configuration Task

Before configuring BGP4+ network security, familiarize yourself with the applicable environment, complete the pre-configuration tasks, and obtain the required data. This can help you complete the configuration task quickly and accurately.

Applicable Environment

- BGP4+ authentication

BGP4+ uses TCP as the transport layer protocol. To enhance BGP4+ security, you can perform the Message Digest 5 (MD5) authentication when TCP connections are created. The MD5 authentication, however, does not authenticate BGP4+ packets. Instead, it sets MD5 authentication passwords for TCP connections, and the authentication is then completed by TCP. If the authentication fails, TCP connections cannot be established.

- BGP4+ GTSM

The Generalized TTL Security Mechanism (GTSM) is used to prevent attacks by using the TTL detection. If an attack simulates BGP4+ packets and sends a large number of packets to a router, an interface through which the router receives the packets directly sends the packets to BGP4+ of the control layer, without checking the validity of the packets. In this manner, routers on the control layer process the packets as valid packets. As a result, the system becomes busy, and CPU usage is high.

In this case, you can configure GTSM to solve the preceding problem. After GTSM is configured on a router, the router checks whether the TTL value in the IP header of a packet is in the pre-defined range after receiving the packet. If yes, the router forwards the packet; if not, the router discards the packet. This enhances the security of the system.

NOTE

- The AR150/200 supports BGP4+ GTSM.
- GTSM supports only unicast addresses; therefore, GTSM needs to be configured on all the routers configured with routing protocols.

Pre-configuration Tasks

Before configuring BGP4+ security, complete the following task:

- [Configuring Basic BGP4+ Functions](#)

Data Preparation

Before configure BGP4+ security, you need the following data.

No.	Data
1	BGP4+ peer address or name of the peer group of each router
2	MD5 authentication password
3	Key-Chain authentication name

9.13.2 Configuring MD5 Authentication

In MD5 authentication of BGP4+, you only need to set MD5 authentication passwords for TCP connections, and the authentication is performed by TCP. If the authentication fails, TCP connections cannot be established.

Context

Do as follows on the BGP4+ router:

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
bgp as-number
```

The BGP view is displayed.

Step 3 Run:

```
peer { ipv6-address | group-name } password { cipher cipher-password | simple  
simple-password }
```

The MD5 authentication password is configured.

 **NOTE**

When this command is used in the BGP4+ view, the extensions on VPNv6 of MP-BGP are also valid because they use the same TCP connection.

Characters ^#^# and @\$@\$ are used to identify passwords with variable lengths. Characters ^#^# are the prefix and suffix of a new password, and characters @\$@\$ are the prefix and suffix of an old password. Neither of them can be both configured at the beginning and end of a plain text password.

The BGP MD5 authentication and BGP Keychain authentication are mutually exclusive.

---End

9.13.3 Configuring Keychain Authentication

You need to configure Keychain authentication on both BGP4+ peers, and ensure that encryption algorithms and passwords configured for Keychain authentication on both peers are the same. Otherwise, TCP connections cannot be established between BGP4+ peers, and BGP4+ messages cannot be exchanged.

Context

Do as follows on the BGP4+ router:

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
bgp as-number
```

The BGP view is displayed.

Step 3 Run:

```
peer { ipv6-address | group-name } keychain keychain-name
```

The Keychain authentication is configured.

You must configure Keychain authentication on both BGP peers. Note that encryption algorithms and passwords configured for the Keychain authentication on both peers must be the same; otherwise, the TCP connection cannot be set up between BGP peers and BGP messages cannot be transmitted.

Before configuring the BGP Keychain authentication, configure a Keychain in accordance with the configured *keychain-name*. Otherwise, the TCP connection cannot be set up.

NOTE

- When this command is used in the BGP view, the extensions on VPNv6 of MP-BGP are also valid because they use the same TCP connection.
- The BGP MD5 authentication and BGP Keychain authentication are mutually exclusive.

---End

9.13.4 Configuring Basic BGP4+ GTSM Functions

The GTSM mechanism protects a router by checking whether the TTL value in the IP header is in a pre-defined range.

Procedure

- Configuring Basic BGP4+ GTSM Functions

Do as follows on the two peers:

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
bgp as-number
```

The BGP view is displayed.

3. Run

```
peer { group-name | ipv6-address } valid-ttl-hops [ hops ]
```

Basic BGP4+ GTSM functions are configured.

The range of TTL values of packets is [255-hops+1, 255]. By default, the value of hops is 255. That is, the valid TTL range is [1, 255]. For example, for the direct EBGP route, the value of hops is 1. That is, the valid TTL value is 255.

 **NOTE**

- The configuration in the BGP view is also valid for the VPNv6 extension of MP-BGP. This is because they use the same TCP connection.
- GSTM is exclusive with EBGP-MAX-HOP; therefore, you can enable only one of them on the same peer or the peer group.

After the BGP4+ GTSM policy is configured, an interface board checks the TTL values of all BGP4+ packets. According to the actual networking requirements, you can configure GTSM to discard or process the packets that do not match the GTSM policy. If you configure GTSM to discard the packets that do not match the GTSM policy by default, you can configure the range of finite TTL values according to the network topology; therefore, the interface board directly discards the packets with the TTL value not in the configured range. Thus, the attackers cannot simulate valid BGP4+ packets to occupy CPU resources.

- Performing the Default GTSM Action

Do as follows on the router configured with GTSM:

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
gtsm default-action { drop | pass }
```

The default action is configured for the packets that do not match the GTSM policy.

By default, the packets that do not match the GTSM policy can pass the filtering.

 **NOTE**

If only the default action is configured and the GTSM policy is not configured, GTSM does not take effect.

- Configuring the Log for Discarded Packets

Do as follows on the router configured with GTSM:

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
gtsm log drop-packet all
```

The log function is enabled on all the slots.

Information that GTSM drops packets is recorded in the log.

The log records information that GTSM drops packets, which helps locate faults.

----End

9.13.5 Checking the Configuration

After BGP4+ network security is configured, you can check authentication information of BGP4+ peers.

Prerequisites

The configurations for BGP4+ security are complete.

Procedure

- Run the **display gtsm statistics all** command to check the statistics of GTSM.

Run the **display gtsm statistics** command. You can view GTSM statistics on each board, including the total number of BGP4+ packets, the total number of OSPF packets, the number of packets that match the GTSM policy, and the number of discarded packets.

- Run the **display bgp ipv6 peer ipv6-address verbose** command to check information about BGP4+ GTSM.
- Run the **display bgp group [group-name]** command to check GTSM of a BGP4+ peer group.

Run the **display bgp peer verbose** and the **display bgp group** commands. You can find the configured maximum valid TTL value and GTSM being enabled on a BGP4+ peer or peer group.

----End

9.14 Maintaining BGP4+

Maintaining BGP4+ involves resetting a BGP4+ connection and clearing BGP4+ statistics.

9.14.1 Resetting BGP4+ Connections

This section describes how to clear the statistics of BGP4+ accounting, flapped routes, and suppressed routes.

Context



CAUTION

The peer relationship is broken after you reset the BGP4+ connections with the **reset bgp ipv6** command. Exercise caution when running this command.

After the BGP4+ configuration changes, reset the BGP4+ connections to validate the modification.

To reset the BGP4+ connections, run the following **reset** command in the user view.

Procedure

- To validate the new configuration, run the **reset bgp ipv6 all** command in the user view to reset all the BGP4+ connections.
- To validate the new configuration, run the **reset bgp ipv6 as-number** command in the user view to reset the BGP+4 connections between the peers in a specified AS.
- To validate the new configuration, run the **reset bgp ipv6 { ipv6-address | group group-name }** command in the user view to reset the BGP+4 connections with the specified peer (or peer group).
- To validate the new configuration, run the **reset bgp ipv6 external** command in the user view to reset the external BGP4+ connections.
- To validate the new configuration, run the **reset bgp ipv6 internal** command in the user view to reset the internal BGP4+ connections.

----End

9.14.2 Clearing BGP4+ Statistics

Devices can generate debugging information after the debugging of a module is enabled in the user view. Debugging information shows the contents of the packets sent or received by the debugged module.

Context



CAUTION

The BGP4+ statistics cannot be restored after being cleared. Exercise caution when running this command.

Procedure

- Run the **reset bgp ipv6 dampening [ipv6-address prefix-length]** command in the user view to clear information about route dampening and release the suppressed routes.
- Run the **reset bgp ipv6 flap-info [ipv6-address prefix-length | regexp as-path-regexp | as-path-filter { as-path-filter-number | as-path-filter-name }]** command in the user view to clear the statistics of route flapping.

----End

9.15 Configuration Examples

BGP4+ configuration examples explain networking requirements, networking diagram, configuration notes, configuration roadmap, and configuration procedure.

9.15.1 Example for Configuring Basic BGP4+ Functions

Before building BGP4+ networks, you need to configure basic BGP4+ functions.

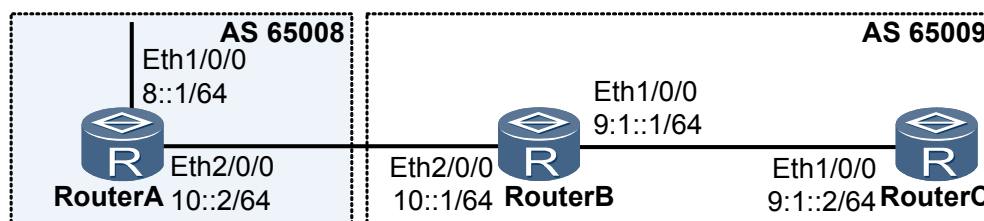
Networking Requirement

As shown in **Figure 9-1**, there are two ASs: 65008 and 65009. Router A belongs to AS 65008; Router B, and Router C belong to AS65009. BGP4+ is required to exchange the routing information between the two ASs.

**NOTE**

AR150/200 is RouterC.

Figure 9-1 Networking diagram of configuring basic BGP4+ functions



Configuration Roadmap

The configuration roadmap is as follows:

1. Configure the IBGP connections among Router B, and Router C.
2. Configure the EBGP connection between Router A and Router B.

Data Preparation

To complete the configuration, you need the following data:

- The router ID of Router A is 1.1.1.1. Its AS number is 65008.
- The router IDs of Router B, and Router C are 2.2.2.2, and 3.3.3.3 respectively. Their AS number is 65009.

Procedure

Step 1 Assign an IPv6 address for each interface.

The details are not mentioned here.

Step 2 Configure the IBGP.

Configure Router B.

```
[RouterB] ipv6
[RouterB] bgp 65009
[RouterB-bgp] router-id 2.2.2.2
[RouterB-bgp] peer 9:1::2 as-number 65009
[RouterB-bgp] ipv6-family unicast
[RouterB-bgp-af-ipv6] peer 9:1::2 enable
[RouterB-bgp-af-ipv6] network 9:1:: 64
```

Configure Router C.

```
[RouterC] ipv6
[RouterC] bgp 65009
```

```
[RouterC-bgp] router-id 3.3.3.3
[RouterC-bgp] peer 9:1::1 as-number 65009
[RouterC-bgp] ipv6-family unicast
[RouterC-bgp-af-ipv6] peer 9:1::1 enable
[RouterC-bgp-af-ipv6] network 9:1:: 64
```

Step 3 Configure the EBGP.

Configure Router A.

```
[RouterA] ipv6
[RouterA] bgp 65008
[RouterA-bgp] router-id 1.1.1.1
[RouterA-bgp] peer 10::1 as-number 65009
[RouterA-bgp] ipv6-family unicast
[RouterA-bgp-af-ipv6] peer 10::1 enable
[RouterA-bgp-af-ipv6] network 10:: 64
[RouterA-bgp-af-ipv6] network 8:: 64
```

Configure Router B.

```
[RouterB] bgp 65009
[RouterB-bgp] peer 10::2 as-number 65008
[RouterB-bgp] ipv6-family unicast
[RouterB-bgp-af-ipv6] peer 10::2 enable
[RouterB-bgp-af-ipv6] network 10:: 64
```

Check the connection status of BGP4+ peers.

```
[RouterB] display bgp ipv6 peer

BGP local router ID : 2.2.2.2
Local AS number : 65009
Total number of peers : 2                Peers in established state : 2

Peer          V          AS  MsgRcvd  MsgSent  OutQ  Up/Down    State
PrefRcv
-----
 9:1::2      4          65009    10       14       0 00:07:10 Established
1
10::2       4          65008     6        6       0 00:02:17 Established
2
```

The routing table shows that Router B has set up BGP4+ connections with other routers.

Display the routing table of Router A.

```
[RouterA] display bgp ipv6 routing-table

BGP Local router ID is 1.1.1.1
Status codes: * - valid, > - best, d - damped,
              h - history, i - internal, s - suppressed, S - Stale
              Origin : i - IGP, e - EGP, ? - incomplete

Total Number of Routes: 4
*> Network    : 8:::                               PrefixLen : 64
   NextHop    : ::                               LocPrf    :
   MED        : 0                                PrefVal    : 0
   Label      :
   Path/Ogn   : i
*> Network    : 9:1:::                             PrefixLen : 64
   NextHop    : 10::1                             LocPrf    :
   MED        : 0                                PrefVal    : 0
   Label      :
   Path/Ogn   : 65009 i
*> Network    : 10:::                               PrefixLen : 64
   NextHop    : ::                               LocPrf    :
   MED        : 0                                PrefVal    : 0
   Label      :
```

```

    Path/Ogn : i

    NextHop  : 10::1
    MED      : 0
    Label    :
    Path/Ogn : 65009 i

    LocPrf   :
    PrefVal  : 0
    
```

The routing table shows that Router A has learned the route from AS 65009. AS 65008 and AS 65009 can exchange their routing information.

----End

Configuration Files

- Configuration file of Router A

```

#
 sysname RouterA
#
 ipv6
#
 interface Ethernet1/0/0
  ipv6 enable
  ipv6 address 8::1/64
#
 interface Ethernet2/0/0
  ipv6 enable
  ipv6 address 10::2/64
#
 bgp 65008
  router-id 1.1.1.1
  peer 10::1 as-number 65009
#
  ipv4-family unicast
   undo synchronization
#
  ipv6-family unicast
   undo synchronization
   network 8:: 64
   network 10:: 64
   peer 10::1 enable
#
 return
    
```

- Configuration file of Router B

```

#
 sysname RouterB
#
 ipv6
#
 interface Ethernet1/0/0
  ipv6 enable
  ipv6 address 9:1::1/64
#
 interface Ethernet2/0/0
  ipv6 enable
  ipv6 address 10::1/64
#
 bgp 65009
  router-id 2.2.2.2
  peer 9:1::2 as-number 65009
  peer 10::2 as-number 65008
#
  ipv4-family unicast
   undo synchronization
#
  ipv6-family unicast
   undo synchronization
   network 9:1:: 64
    
```

```
network 10:: 64
peer 9:1::2 enable
peer 10::2 enable
#
return
```

● Configuration file of Router C

```
#
sysname RouterC
#
ipv6
#
interface Ethernet1/0/0
ipv6 enable
ipv6 address 9:1::2/64
#
bgp 65009
router-id 3.3.3.3
peer 9:1::1 as-number 65009
#
ipv4-family unicast
undo synchronization
#
ipv6-family unicast
undo synchronization
network 9:1:: 64
peer 9:1::1 enable
#
return
```

10 Routing Policy Configuration

About This Chapter

Routing policies are used to filter routes to change the path through which network traffic passes.

[10.1 Overview of the Routing Policy](#)

By using routing policies, you can flexibly control the routes to be sent or received.

[10.2 Routing Policy Features Supported by the AR150/200](#)

When configuring routing policies, you can use these filters: ACL, IP prefix list, AS-Path filter, community filter, extended community filter, RD filter, and Route-Policy.

[10.3 Configuring the IP-Prefix List](#)

An IP prefix list filters routes according to the destination addresses of the routes.

[10.4 Configuring the Route-Policy](#)

Each node of a Route-Policy consists of a set of if-match and apply clauses.

[10.5 Applying Filters to Received Routes](#)

By applying the related filters of routing policies to routing protocols, you can filter the received routes.

[10.6 Applying Filters to Advertised Routes](#)

By applying the related filters of routing policies to routing protocols, you can filter advertised routes.

[10.7 Applying Filters to Imported Routes](#)

By applying the related filters of routing policies to routing protocols, you can filter imported routes.

[10.8 Controlling the Valid Time of the Routing policy](#)

To ensure network stability, you need to configure the delay for applying a routing policy when modifying the routing policy.

[10.9 Maintaining the Routing Policy](#)

Maintaining routing policies involves clearing the statistics of the IP prefix list and debugging routing policies.

[10.10 Configuration Examples](#)

The configuration examples in this section explain the networking requirements, networking diagram, configuration notes, configuration roadmap, and configuration procedure for different types of routing policies.

10.1 Overview of the Routing Policy

By using routing policies, you can flexibly control the routes to be sent or received.

Routing Policy

Routing policies are used to filter routes and control the receiving and advertising of routes. By changing the route attributes such as reachability, you can change the path that the traffic passes through.

When a router sends or receives routes, it may use certain policies to filter routes. The policies are used in the following situations:

- Send or receive routes that meet the matching rules.
- A routing protocol such as the Routing Information Protocol (RIP) needs to import the routes discovered by other routing protocols to enrich its routing information. When importing routes from other routing protocols, the router may import certain routes that meet the matching rules, and set attributes of the routes imported to meet the requirement.

To implement a routing policy, you must:

- Define a set of matching rules and setting rules. The policy is applied to the routing information to meet the requirements of the matching rules.
- Apply the matching rules to the routing policies for route advertisement, reception, and import.

Differences Between Routing Policy and PBR

Different from the forwarding by searching the Forwarding information base (FIB) according to the destination address of a packet, Policy-based routing (PBR) is a route selection mechanism based on policies set by users. PBR supports the information based on the source address and the length of a packet. PBR selects routes according to the set policy. PBR can be applicable to security and load balancing.

Routing policies and PBR are different concepts. [Table 10-1](#) shows the differences between the two concepts.

Table 10-1 Differences between routing policy and PBR

Routing policy	Policy-based routing
Forwards packets based on the destination address in the routing table.	Forwards packets based on the policy. If packets fail to be forwarded, the device forwards packets by searching the routing table.
Based on the control plane and serves the routing protocol and routing table.	Based on forwarding plane and serves for the forwarding policy.
Combines with the routing protocol	Needs to be manually configured hop by hop to ensure that the packet is forwarded through the policy.
The route-policy command is used.	The policy-based-route command is used.

10.2 Routing Policy Features Supported by the AR150/200

When configuring routing policies, you can use these filters: ACL, IP prefix list, AS-Path filter, community filter, extended community filter, RD filter, and Route-Policy.

Filters

The AR150/200 provides several types of filters for routing protocols, such as Access Control Lists (ACLs), IP prefix lists, AS-Path filters, community filters, extended community filters (Extcommunity-filters), and Route-Policies.

- **ACL**

The ACL consists of the ACL for IPv4 packets. According to the usage, ACLs are classified into three types, that is, interface-based ACLs, basic ACLs, and advanced ACLs. When defining an ACL, you can specify the IP address and subnet range to match the destination network segment address or the next hop address of a route.

For details of the ACL configuration, refer to the Huawei AR150&200 Series Enterprise Routers *Configuration Guide - IP Services*.
- **IP-Prefix List**

The IP-prefix list consists of IPv4 prefix list and IPv6 prefix list. The implementation of the IP-prefix is flexible.

An IP-prefix list is identified by its list name. Each prefix list includes multiple entries. Each entry can independently specify the matching range in the form of the network prefix. The matching range is identified by an index number that designates the sequence of the matching check.

During the matching, the router checks entries identified by index numbers in an ascending order. When a route matches an entry, the system does not search the next entry matching the route. For the detailed configuration, refer to [Configuring the IP-Prefix List](#).
- **AS-Path Filter**

Border Gateway Protocol (BGP) routing information packet includes an autonomous system (AS) path domain. The AS-Path filter specifies the matching condition for the AS path domain.

For the configuration of AS-Path filter, refer to .
- **Community Filter**

The community filter is used only in BGP. The BGP routing information includes a community attribute domain. It is used to identify a community. The community filter specifies the matching condition for the community attribute domain.

For the configuration of community filter, refer to [BGP Configuration](#).
- **Extcommunity-Filter**

The Extcommunity-filter is used only in BGP. The extended community of BGP supports only the Router-Target (RT) extended community of Virtual Private Network (VPN). The Extcommunity-filter specifies matching rules for the extended community attribute.

For the configuration of excommunity-filter, refer to [BGP Configuration](#).
- **RD Filter**

Through Route Distinguisher (RD), the VPN instance implements the independency of address space and identifies the IPv4 and IPv6 prefixes of the same address space. The RD attribute filter specifies matching conditions for different RDs.

For the configuration of the RD attribute filter, refer to the Huawei AR150&200 Series Enterprise Routers *Configuration Guide - VPN*.

- Route-Policy

The Route-Policy is a complex filter. A Route-Policy is used to match certain route attributes, and to change the route attributes when certain matching rules are met. The Route-Policy uses the preceding filters to define its filtering rules.

A Route-Policy consists of multiple nodes. The relationship between the nodes is "OR". The system checks the nodes in the routing policy, the node with the smaller value of node is checked first. When the route matches a node in the routing policy, it passes the Route-Policy and the system does not search the next matching node.

Each node comprises a set of **if-match** and **apply** clauses. The **if-match** clauses define the matching rules. The matching objects are certain route attributes. The relationship between **if-match** clauses in a node is "AND". A matching succeeds only when all the matching rules specified by the **if-match** clauses in the same node are matched.

The **apply** clauses specify actions. When a route matches a rule, the **apply** clause sets certain attributes for the route. For the detailed configuration, refer to [Configuring the Route-Policy](#).

Application of the Routing Policy

The routing policy is used in the following situations:

- Import routes that meet the matching rules through filters when a routing protocol imports routes discovered by other protocols.
- Filter routes that a routing protocol advertises or receives. Only the routes that meet the matching rules are received or advertised.

For the configuration of routing policy applications, refer to the related routing protocol configurations.

 **NOTE**

After the routing policy changes, Routing Management Module (RM) immediately notifies various protocols for processing by default.

10.3 Configuring the IP-Prefix List

An IP prefix list filters routes according to the destination addresses of the routes.

10.3.1 Establishing the Configuration Task

Before configuring the IP prefix list, familiarize yourself with the applicable environment, complete the pre-configuration tasks, and obtain the required data. This can help you complete the configuration task quickly and accurately.

Applicable Environment

Before applying a routing policy, you should set the matching rules, that is, filters. Compared with an ACL, an IP prefix list is more flexible. When the IP prefix list is used to filter routes, it matches the destination address of a route.

Pre-configuration Tasks

None.

Data Preparation

To configure an IP prefix list, you need the following data.

No.	Data
1	Name of IP prefix list
2	Matched address range

10.3.2 Configuring an IPv4 Prefix List

An IP prefix list filters routes according to IP address prefixes. An IP address prefix is defined by the IP address and mask length.

Context

Do as follows on the router to which the IP prefix list is applied:

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
ip ip-prefix ip-prefix-name [ index index-number ] { permit | deny } ip-address  
mask-length [ greater-equal greater-equal-value ] [ less-equal less-equal-value ]
```

An IPv4 prefix list is configured.

The range of the mask length can be specified as $mask-length \leq greater-equal-value \leq less-equal-value \leq 32$. If only **greater-equal** is specified, the range of the prefix is [*greater-equal-value*, 32]; if only **less-equal** is specified, the range of the prefix is [*mask-length*, *less-equal-value*].

An IPv4 prefix list is identified by its list name. Each prefix list contains multiple entries. Each entry can independently specify the matching range in the form of the network prefix and identify it with an index number. For example, the following shows an IPv4 prefix list named **abcd**:

```
#  
ip ip-prefix abcd index 10 permit 1.0.0.0 8  
ip ip-prefix abcd index 20 permit 2.0.0.0 8
```

During the matching, the system checks the entries identified by the index numbers in an ascending order. When a route matches an entry, it does not match other entries.

In the AR150/200, all unmatched routes cannot pass the filtering list. If all entries are in **deny** mode, all routes are filtered. It is recommended that you define a **permit 0.0.0.0 0 less-equal 32** entry after multiple entries in **deny** mode, thus allowing all the other IPv4 routes to pass the IP prefix list.

 **NOTE**

If more than one IP-prefix entry is defined, at least one entry should be in the **permit** mode.

----End

10.3.3 Configuring an IPv6 Prefix List

An IPv6 prefix list filters routes according to IPv6 address prefixes. An IPv6 address prefix is defined by the IPv6 address and mask length.

Context

Do as follows on the router to which the IP prefix list is applied:

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
ip ipv6-prefix ipv6-prefix-name [ index index-number ] { permit | deny } ipv6-  
address prefix-length [ greater-equal greater-equal-value ] [ less-equal less-  
equal-value ]
```

An IPv6 prefix list is configured.

The range of the prefix length can be specified as $prefix-length \leq greater-equal-value \leq less-equal-value \leq 128$. If only **greater-equal** is specified, the range of the prefix is [*greater-equal-value*, 128]; if only **less-equal** is specified, the range of the prefix is [*prefix-length*, *less-equal-value*].

An IPv6 prefix list is identified by its list name. Each prefix list can include multiple entries. Each entry can independently specify the matching range in the form of the network prefix and identify it with an index number. For example, the following shows an IPv6 prefix list named **abcd**:

```
#  
ip ipv6-prefix abcd index 10 permit 1:: 64  
ip ipv6-prefix abcd index 20 permit 2:: 64
```

During the matching, the system checks the entries identified by the index numbers in an ascending order. When a route matches an entry, it does not match other entries.

In AR150/200, all unmatched routes are filtered. If all entries are in **deny** mode, all routes are filtered. It is recommended that you define a **permit :: 0 less-equal 128** after multiple entries in **deny** mode, thus allowing all the other IPv6 routes to pass the IP prefix list.

 **NOTE**

If more than one IP-prefix entry is defined, at least one entry should be in the **permit** mode.

----End

10.3.4 Checking the Configuration

After an IP prefix list is configured, you can check information about the IP prefix list.

Prerequisites

The configurations for the IP-Prefix list are complete.

Procedure

- Run the **display ip ip-prefix** [*ip-prefix-name*] command to check information about the IPv4 prefix list.
- Run the **display ip ipv6-prefix** [*ipv6-prefix-name*] command to check information about the IPv6 prefix list.

----End

10.4 Configuring the Route-Policy

Each node of a Route-Policy consists of a set of if-match and apply clauses.

10.4.1 Establishing the Configuration Task

Before configuring the Route-Policy, familiarize yourself with the applicable environment, complete the pre-configuration tasks, and obtain the required data. This can help you complete the configuration task quickly and accurately.

Applicable Environment

A Route-Policy is used to match routes or certain route attributes, and to change these attributes when the matching rules are met.

A Route-Policy consists of multiple nodes. Each node is classified into the following clauses:

- **if-match** clauses: define the matching rules. The matching rules are used by the routes that match the Route-Policy. The matching objects refer to some attributes of the route.
- **apply** clauses: specify actions, that is, configuration commands used to modify certain attributes.

For more information about Route-Policy, refer to the *Huawei AR150&200 Series Enterprise Routers Feature Description - IP Routing*.

Pre-configuration Tasks

To configure a Route-Policy, complete the following tasks:

- [10.3 Configuring the IP-Prefix List](#)
- Configuring routing protocols

Data Preparation

To configure a Route-Policy, you need the following data.

No.	Data
1	Name and node number of the Route-Policy
2	Matching rule
3	Route attributes to be modified

10.4.2 Creating a Route-Policy

By applying a Route-Policy, you can set attributes for the imported routes according to networking requirements.

Context

Do as follows on the router to which the Route-Policy is applied:

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
route-policy route-policy-name { permit | deny } node node
```

A node of the Route-Policy is created and the Route-Policy view is displayed.

- The parameter **permit** specifies a node in a Route-Policy in **permit** mode. If a route matches the node, the router performs actions defined by the **apply** clauses and the matching is complete. Otherwise, the route continues to match the next node.
- The parameter **deny** specifies a node in a Route-Policy in **deny** mode. In **deny** mode, the **apply** clauses are not used. If a route entry matches all the **if-match** clauses of the node, the route is denied by the node and the next node is not matched. If the entry does not match all the clauses, the next node is matched.

 **NOTE**

In the AR150/200, by default, the unmatched routes are denied. If multiple nodes are defined in a Route-Policy, at least one of them should be in **permit** mode.

When the parameter **route-policy** is used to filter routes, note the following:

- If a route does not match any node, it is denied by the Route-Policy.
- If all the nodes in the routing policy are in **deny** mode, all the routes are denied by the Route-Policy.

When a Route-Policy is used to filter the routing information, the node with the smaller value of *node* is tested first.

Step 3 (Optional) Run:

```
description text
```

The description of the routing policy is configured.

----End

10.4.3 (Optional) Configuring the If-Match Clause

The **if-match** clauses define the rules for matching certain route attributes.

Context

Do as follows on the router to which the Route-Policy is applied:

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
route-policy route-policy-name { permit | deny } node node
```

The Route-Policy view is displayed.

Step 3 Run the following command as required:

● Run:

```
if-match acl { acl-number | acl-name }
```

The ACL is configured to match the routes.

● Run:

```
if-match cost cost
```

The cost is set to match the routes.

● Run:

```
if-match interface interface-type interface-number
```

The outbound interface is configured to match the routes.

● Run:

```
if-match ip { next-hop | route-source | group-address } { acl { acl-number | acl-name } | ip-prefix ip-prefix-name }
```

The next hop, the source address or the multicast group address is configured to match the routes.

● Run:

```
if-match ip-prefix ip-prefix-name
```

The IP prefix list is configured to match the routes.

 **NOTE**

For the same Route-Policy node, you cannot run the **if-match acl** command and the **if-match ip-prefix** command at the same time. This is because the latest configuration overrides the previous configuration.

● Run:

```
if-match ipv6 { address | next-hop | route-source } prefix-list ipv6-prefix-name
```

The next hop or the source address is configured to match the routes.

- Perform as follows to match the type of the route:

- Run:

```
if-match route-type { external-type1 | external-type1or2 | external-type2 |  
internal | nssa-external-type1 | nssa-external-type1or2 | nssa-external-  
type2 }
```

The route type, OSPF in this case, is set to match the routes.

- Run:

```
if-match route-type { is-is-level-1 | is-is-level-2 }
```

The route type, IS-IS in this case, is set to match the routes.

- Run:

```
if-match tag tag
```

The tag is set to match the routes.

The commands in Step 3 can be used regardless of the order. A node can have multiple or no **if-match** clauses.

NOTE

- For the same node in a route-policy, the relationship between **if-match** clauses is "AND". The route must meet all the matching rules before the actions defined by the **apply** clauses are performed. In the **if-match route-type** and **if-match interface** commands, the relationship between the **if-match** clauses is "OR". In other commands, the relationship between **if-match** clauses is "AND".
- If no **if-match** clause is specified, all the routes meet the matching rules.

---End

10.4.4 (Optional) Configuring the Apply Clause

The **apply** clauses specify actions to set certain route attributes.

Context

Do as follows on the router to which the Route-Policy is applied:

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
route-policy route-policy-name { permit | deny } node node
```

The Route-Policy view is displayed.

Step 3 Run the following command as required:

- Run:

```
apply cost [ + | - ] cost
```

The cost of the route is set.

- Set the cost type of the route.

- Run the **apply cost-type** { **external** | **internal** } command to set the cost type of an IS-IS route.
- Run the **apply cost-type** { **type-1** | **type-2** } command to set the cost type of an OSPF route.

The **apply cost-type** { **external** | **internal** } command and the **apply cost-type** { **type-1** | **type-2** } command are mutually exclusive and cannot be configured at the same time.

- Run:

```
apply ip-address next-hop { peer-address | ipv4-address }
```

The next hop address of the IPv4 route is set.

- Run:

```
apply ipv6 next-hop { peer-address | ipv6-address }
```

The next hop address of the IPv6 route is set.

- Run:

```
apply isis { level-1 | level-1-2 | level-2 }
```

The route level of IS-IS is set.

- Run:

```
apply ospf { backbone | stub-area }
```

The area of the OSPF that routes are imported into is set.

- Run:

```
apply preference preference
```

The preference of the routing protocol is set.

The smaller the preference value, the higher the preference.

- Run:

```
apply tag tag
```

The tag of the route is set.

The commands in Step 3 can be used regardless of the order.

----End

10.4.5 Checking the Configuration

After the Route-Policy is configured, you can check information about the Route-Policy.

Prerequisites

The configurations for the Route-Policy are complete.

Procedure

- Run the **display route-policy** [*route-policy-name*] command to check the Route-Policy.

----End

10.5 Applying Filters to Received Routes

By applying the related filters of routing policies to routing protocols, you can filter the received routes.

10.5.1 Establishing the Configuration Task

Before applying filters to the received routes, familiarize yourself with the applicable environment, complete the pre-configuration tasks, and obtain the required data. This can help you complete the configuration task quickly and accurately.

Applicable Environment

After defining the filters including the IP prefix list, ACL, and Route-Policy related to the routing policy, you need to import the filters to the protocols. The routing filters are used in the following situations:

- Filtering the received routes

Use the **filter-policy** command in the protocol view and apply an ACL or an IP prefix list to filter the received routes. Only the routes that meet the matching rules are received.

The **filter-policy import** command is used to filter the received routes.

For the distance vector (DV) protocol and the link state protocol, the procedures are different after the **filter-policy** command is run.

- DV protocol

A DV protocol generates routes based on the routing table. The filters affect the routes received from the neighbor and the routes to be sent to the neighbor.

- Link state protocol

A link state protocol generates routes based on the Link State Database. The **filter-policy** command does not affect the Link State Advertisements (LSAs) or the integrity of the LSDB. Therefore, the effect on the commands of **filter-policy import** and **filter-policy export** are different.

The **filter-policy import** command identifies the route that is added to a local routing table from a protocol routing table only. That is, this command affects the local routing table only, but does not affect the protocol routing table.

 NOTE

- BGP has powerful filtering functions. For details of BGP configuration, refer to [BGP Configuration](#).
- You can run the **filter-policy** command and the **import-route** command with different parameters for RIP, OSPF, IS-IS, and BGP. For details, refer to related configurations.

Pre-configuration Tasks

Before applying filters to received routes, complete the following tasks:

- [Configuring the IP-Prefix List](#)
- Configuring an ACL
- [Configuring the Route-Policy](#)

Data Preparation

To apply filters to received routes, you need the following data.

No.	Data
1	Name of the IP prefix list
2	Name of the ACL
3	Name of the Route-Policy and node number

10.5.2 Filtering Routes Received by RIP

By applying filters, you can control the receiving of RIP routes.

Context

Do as follows on the router that runs RIP:

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
rip [ process-id ] [ vpn-instance vpn-instance-name ]
```

Step 3 Run either of the following commands as required:

- **filter-policy** { *acl-number* | **acl-name** *acl-name* } **import** [*interface-type interface-number*]
- **filter-policy gateway** *ip-prefix-name* **import**
- **filter-policy ip-prefix** *ip-prefix-name* [**gateway** *ip-prefix-name*] **import** [*interface-type interface-number*]

The filtering policy is configured for routes received by RIP.

The **filter-policy** is configured in the RIP process. If routes are filtered based on an interface, you can configure only one route-policy based on the interface at a time. If no interface is specified, the system considers the configured route-policy as the global route-policy, and you can configure only one route-policy at a time. If the route-policy is configured repeatedly, the new route-policy will replace the old route-policy.

---End

10.5.3 Filtering Routes Received by OSPF

By applying filters, you can control the receiving of OSPF routes.

Context

Do as follows on the router that runs OSPF:

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
ospf [ process-id ]
```

An OSPF process is enabled and the OSPF view is displayed.

Step 3 Run:

```
filter-policy { acl-number | acl-name acl-name | ip-prefix ip-prefix-name } import
```

The filtering policy is configured for routes received by OSPF.

----End

10.5.4 Filtering Routes Received by IS-IS

By applying filters, you can control the receiving of IS-IS routes.

Context

Do as follows on the router that runs IS-IS:

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
isis [ process-id ] [ vpn-instance vpn-instance-name ]
```

An IS-IS process is enabled and the IS-IS view is displayed.

Step 3 Run:

```
filter-policy { acl-number | acl-name acl-name | ip-prefix ip-prefix-name | route-policy route-policy-name } import
```

You can configure IS-IS to filter the received routes to be added to the IP routing table.

----End

10.5.5 Filtering Routes Received by BGP

By applying filters, you can control the receiving of BGP routes.

Procedure

- Filtering the Received Routes

Do as follows on the router that runs BGP:

1. Run:

```
system-view
```

The system view is displayed.

2. Run:
`bgp as-number`

The BGP view is displayed.

3. Run:
`filter-policy { acl-number | acl-name acl-name | ip-prefix ip-prefix-name } import`

The filtering policy is configured for routes received by BGP.

- Filtering Routes Received from the Peers or Peer Groups

Do as follows on the router that runs BGP:

1. Run:
`system-view`
The system view is displayed.
2. Run:
`bgp as-number`
The BGP view is displayed.
3. Run:
`ipv4-family unicast`

The IPv4 unicast address family view is displayed.

4. Run:
`peer { group-name | ipv4-address } filter-policy { acl-number | acl-name acl-name } import`

The filtering policy is configured for routes received from peers or peer groups.

----End

10.5.6 Checking the Configuration

After filters are applied to the received routes, you can check information about the routing table of each protocol.

Prerequisites

The configurations for applying filters to received routes are complete.

Procedure

- Run the **display rip process-id route** command to check information about the RIP routing table.
- Run the **display ospf [process-id] routing** command to check information about the OSPF routing table.
- Run the **display isis [process-id] route** command to check information about the ISIS routing table.
- Run the **display bgp routing-table** command to check information about the BGP routing table.
- Run the **display ip routing-table** command to check information about the public IPv4 routing table.

Run the **display ip routing-table** command on the neighboring router. You can find that the routes that meet the matching rules set on the neighboring router are filtered or the actions defined by the **apply** clauses are performed.

----End

10.6 Applying Filters to Advertised Routes

By applying the related filters of routing policies to routing protocols, you can filter advertised routes.

10.6.1 Establishing the Configuration Task

Before applying filters to advertised routes, familiarize yourself with the applicable environment, complete the pre-configuration tasks, and obtain the required data. This can help you complete the configuration task quickly and accurately.

Applicable Environment

After defining the filters including the IP prefix list, ACL, and Route-Policy related to the routing policy, you need to import the filters to the protocols.

- Filtering the advertised routes

Use the **filter-policy** command in the protocol view and import an ACL or an IP prefix list to filter the advertised routes. Only the routes that meet the matching rules are advertised.

The **filter-policy export** command is used to filter the advertised routes.

For the DV protocol and the link state protocol, the procedures are different after the **filter-policy** command is run.

- DV protocol

A DV protocol generates routes based on the routing table. The filters affect the route received from the neighbor and the route to be sent to the neighbor.

- Link state protocol

A link state protocol generates routes based on LSDBs. The **filter-policy** does not affect LSAs or the integrity of LSDBs. The commands of **filter-policy import** and **filter-policy export** are different.

To advertise routes, you can run the **filter-policy export** command on a device to control whether the device advertises the routes imported by a specific routing protocol (such as RIP) from other routing protocols. If the device has not imported any route in **Import** mode, it will not add LSAs or LSPs corresponding to the imported routes to its LSDB. The device, however, can advertise LSAs that carry the routing information discovered by the specific routing protocol itself to other routers.

 **NOTE**

- BGP has powerful filtering function. For details of BGP configuration, refer to [BGP Configuration](#).
- You can run the **filter-policy** command and the **import-route** command with different parameters for RIP, OSPF, IS-IS, and BGP. For details, refer to related configurations.

Pre-configuration Tasks

Before applying filters to advertised routes, complete the following tasks:

- [10.3 Configuring the IP-Prefix List](#)
- Configuring an ACL
- [10.4 Configuring the Route-Policy](#)

Data Preparation

To apply filters to advertised routes, you need the following data.

No.	Data
1	Name of the IP prefix list
2	Name of the ACL
3	Name of the Route-Policy and node number

10.6.2 Filtering Routes Advertised by RIP

By applying filters, you can control the advertisement of RIP routes.

Context

Do as follows on the router that runs RIP:

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
rip [ process-id ]
```

A RIP process is enabled and the RIP view is displayed.

Step 3 Run

```
filter-policy { acl-number | acl-name acl-name | ip-prefix ip-prefix-name } export  
[ protocol [ process-id ] | interface-type interface-number ]
```

The filtering policy is configured for routes advertised by RIP.

The **filter-policy** is configured in the RIP process. If routes are filtered based on an interface, you can configure only one route-policy based on the interface at a time. If no interface is specified, the system considers the configured route-policy as the global route-policy, and you can configure only one route-policy at a time. If the route-policy is configured repeatedly, the new route-policy will replace the old route-policy.

----End

10.6.3 Filtering Routes Advertised by OSPF

By applying filters, you can control the advertisement of OSPF routes.

Context

Do as follows on the router that runs OSPF:

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
ospf [ process-id ]
```

An OSPF process is enabled and the OSPF view is displayed.

Step 3 Run:

```
filter-policy { acl-number | acl-name acl-name | ip-prefix ip-prefix-name } export  
[ protocol [ process-id ] ]
```

The filtering policy is configured to filter the imported routes when these routes are advertised by OSPF.

----End

10.6.4 Filtering Routes Advertised by IS-IS

By applying filters, you can control the advertisement of IS-IS routes.

Context

Do as follows on the router that runs IS-IS:

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
isis [ process-id ]
```

An IS-IS process is enabled and the IS-IS view is displayed.

Step 3 Run:

```
filter-policy { acl-number | acl-name acl-name | ip-prefix ip-prefix-name | route-  
policy route-policy-name } export [ protocol [ process-id ] ]
```

The filtering policy is configured for routes advertised by IS-IS.

----End

10.6.5 Filtering Routes Advertised by BGP

By applying filters, you can control the advertisement of BGP routes.

Procedure

- Filtering the Advertised Routes

Do as follows on the router that runs BGP:

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
bgp as-number
```

The BGP view is displayed.

3. Run:

```
ipv4-family unicast
```

The IPv4 unicast address family view is displayed.

4. Run:

```
filter-policy { acl-number | acl-name acl-name | ip-prefix ip-prefix-name } export [ protocol [ process-id ] ]
```

The filtering policy is configured for routes advertised by BGP.

For the routes imported by BGP, only the routes that meet matching rules can be added to the BGP local routing table and advertised to the BGP peers.

- If *protocol* is specified, only the routes of the specified protocol are filtered.
- If the parameter is not specified, all the routes advertised by BGP are filtered, including the imported routes and the local routes advertised through the **network** command.

 **NOTE**

The **filter-policy export** command of different protocols have different affect ranges on routes advertisement:

- For the link state protocol, only the routes imported are filtered.
- For the DV protocol, the routes imported and the routes discovered by the protocols are filtered.

- Filtering Routes Advertised to the Peers

Do as follows on the router that runs BGP:

1. Run:

```
system-view
```

The system view is displayed.

2. Run:

```
bgp as-number
```

The BGP view is displayed.

3. Run:

```
ipv4-family unicast
```

The IPv4 unicast address family view is displayed.

4. Run:

```
peer { group-name | ipv4-address } filter-policy { acl-number | acl-name acl-name } export
```


The filtering policy is configured for routes advertised to the peers or peer groups.

---End

10.6.6 Checking the Configuration

After filters are applied to advertised routes, you can check information about the routing table of each protocol.

Prerequisites

The configurations for applying filters to advertised routes are complete.

Procedure

- Run the **display rip *process-id* route** command to check information about the RIP routing table.
- Run the **display ospf [*process-id*] routing** command to check information about the OSPF routing table.
- Run the **display isis [*process-id*] route** command to check information about the ISIS routing table.
- Run the **display bgp routing-table** command to check information about the BGP routing table.
- Run the **display ip routing-table** command to check information about the public IPv4 routing table.

Run the **display ip routing-table** command on the neighboring router. You can find that the routes that meets the matching rules set on the neighboring router are filtered or the actions defined by the **apply** clauses are performed.

---End

10.7 Applying Filters to Imported Routes

By applying the related filters of routing policies to routing protocols, you can filter imported routes.

10.7.1 Establishing the Configuration Task

Before applying filters to imported routes, familiarize yourself with the applicable environment, complete the pre-configuration tasks, and obtain the required data. This can help you complete the configuration task quickly and accurately.

Applicable Environment

After defining the filters including the IP prefix list, ACL, and Route-Policy related to the routing policy, you need to import the filters to the protocols.

- Applying the policy to import external routes
 - Use the **import-route** command in the protocol view. Import the required external routes to the protocols and apply a Route-Policy to the imported routes.

- After the external routes are imported, run the **filter-policy export** to filter the routes. Only the routes that meet the matching rules are advertised.

 **NOTE**

- BGP has powerful filtering functions. For details of BGP configuration, refer to [BGP Configuration](#).
- You can run the **filter-policy** command and the **import-route** command with different parameters for RIP, OSPF, IS-IS, and BGP. For details, refer to related configurations.

Pre-configuration Tasks

Before applying filters to imported routes, complete the following tasks:

- [Configuring the IP-Prefix List](#)
- Configuring an ACL
- [Configuring the Route-Policy](#)

Data Preparation

To apply filters to imported routes, you need the following data.

No.	Data
1	Name of the IP prefix list
2	Name of the ACL
3	Name of the Route-Policy and node number

10.7.2 Applying Route-Policy to Routes Imported by RIP

By applying filters, you can control the import of RIP routes.

Context

Do as follows on the router that runs RIP:

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
rip [ process-id ]
```

A RIP routing process is enabled and the RIP view is displayed.

Step 3 Run:

```
import-route bgp [ cost { cost | transparent } | route-policy route-policy-name ]
* or import-route { { static | direct | unr } | { { rip | ospf | isis } [ process-id ] } } [ cost cost | route-policy route-policy-name ] *
```

The external routes are imported.

----End

10.7.3 Applying Route-Policy to Routes Imported by OSPF

By applying filters, you can control the import of OSPF routes.

Context

Do as follows on the router that runs OSPF:

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
ospf [ process-id ]
```

An OSPF process is enabled and the OSPF view is displayed.

Step 3 Run:

```
import-route { limit limit-number | { bgp [ permit-ibgp ] | direct | unr | rip  
[ process-id-rip ] | static | isis [ process-id-isis ] | ospf [ process-id-ospf ] }  
[ cost cost | type type | tag tag | route-policy route-policy-name ] * }
```

The external routes are imported.

----End

10.7.4 Applying Route-Policy to Routes Imported by IS-IS

By applying filters, you can control the import of IS-IS routes.

Context

Do as follows on the router that runs IS-IS:

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
isis [ process-id ]
```

An IS-IS process is enabled and the IS-IS view is displayed.

Step 3 Configuring IS-IS to Import External Routes

- If you want to set the cost for the imported route, you can run the **import-route protocol** [*process-id*] [**cost-type** { **external** | **internal** } | **cost cost** | **tag tag** | **route-policy route-policy-name** | [**level-1** | **level-2** | **level-1-2**]] * command to import the external routes.
- If you want to keep the original cost for the imported route, you can run the **import-route** { { **rip** | **isis** | **ospf** } [*process-id*] | **bgp** } **inherit-cost** [**tag tag** | **route-policy route-policy-name** | [**level-1** | **level-2** | **level-1-2**]] * command to import the external routes.

----End

10.7.5 Applying Route-Policy to Routes Imported by BGP

By applying filters, you can control the import of BGP routes.

Context

Do as follows on the router that runs BGP:

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
bgp as-number
```

The BGP view is displayed.

Step 3 Run:

```
ipv4-family unicast
```

The IPv4 unicast address family view is displayed.

Step 4 Run:

```
import-route protocol [ process-id ] [ med med | route-policy route-policy-name ] *
```

The external routes are imported.

----End

10.7.6 Checking the Configuration

After filters are applied to imported routes, you can check information about the routing table of each protocol.

Prerequisites

The configurations for applying filters to imported routes are complete.

Procedure

- Run the **display rip process-id route** command to check information about the RIP routing table.

- Run the **display ospf [process-id] routing** command to check information about the OSPF routing table.
- Run the **display isis [process-id] route** command to check information about the ISIS routing table.
- Run the **display bgp routing-table** command to check information about the BGP routing table.
- Run the **display ip routing-table** command to check information about the public IPv4 routing table.

Run the **display ip routing-table** command on the neighboring router. You can find that the routes that meet the matching rules on the neighboring router are filtered or the actions defined by the **apply** clauses are performed.

---End

10.8 Controlling the Valid Time of the Routing policy

To ensure network stability, you need to configure the delay for applying a routing policy when modifying the routing policy.

10.8.1 Establishing the Configuration Task

Before configuring the delay for applying a routing policy, familiarize yourself with the applicable environment, complete the pre-configuration tasks, and obtain the required data. This can help you complete the configuration task quickly and accurately.

Applicable Environment

In actual applications, when the configurations of multiple cooperative routing policies change, the Routing Management Module (RM) immediately notifies related protocols to apply a new routing policy, after the configuration of the routing policy is complete. An incomplete routing policy causes route flapping, instability of the network, and a waste of time during packet processing.

The AR150/200 provides the following rules for processing changes of a routing policy:

- By default, the RM immediately notifies the protocol of applying the new policy when the routing policy changes.
- If the valid time of the routing policy is configured, when the commands used to configure the routing policy change, the RM does not notify various protocols of immediately processing the changes. Instead, the RM waits for a certain period, and then notifies various protocols of applying the changed routing policy.
- If the routing policy changes again during the waiting time, the RM resets the timer.

You can run related commands to set the waiting time as required.

Pre-configuration Tasks

None.

Data Preparation

To configure the valid time of the routing policy, you need the following data.

No.	Data
1	Delay for applying the routing policy

10.8.2 Configuring the Delay for Applying the Routing Policy

When modifying multiple cooperative routing policies, you need to configure the delay for applying a routing policy.

Context

Do as follows on the router on which the delay for applying routing policy needs to be changed:

Procedure

Step 1 Run:

```
system-view
```

The system view is displayed.

Step 2 Run:

```
route-policy-change notify-delay delay-time
```

The delay for applying the routing policy is set.

The delay ranges from 1 to 180, in seconds.

By default, the RM immediately notifies the protocol of applying the new policy when the routing policy changes.

Step 3 Run:

```
quit
```

Back to the user view.

Step 4 (Optional) Run:

```
refresh bgp all { export | import }
```

BGP is configured to apply the new routing policy.

After the command is used, policy filtering is immediately effective. You can run the command to configure BGP to immediately apply new policies.

The policies affected by the timer are ACLs, IP prefix lists, AS-Path filters, community filters, extended community filters, RD filters, and Route-Policies.

----End

10.8.3 Checking the Configuration

After the delay for applying a routing policy is configured, you can check the configuration.

Prerequisites

The configurations for controlling the valid time of the routing policy are complete.

Procedure

- Run the **display current-configuration | include notify-delay** command to check the delay for applying the routing policy.

----End

Example

Run the **display current-configuration** command. You can find the delay for applying the routing policy. For example:

```
<Huawei> display current-configuration | include notify-delay  
route-policy-change notify-delay 10
```

10.9 Maintaining the Routing Policy

Maintaining routing policies involves clearing the statistics of the IP prefix list and debugging routing policies.

Context

By default, the statistics of IP prefix lists are not cleared.

Procedure

- Run **reset ip ip-prefix [ip-prefix-name]** command in the user view to clear the IPv4 prefix list statistics.
- Run **reset ip ipv6-prefix [ipv6-prefix-name]** command in the user view to clear the IPv6 prefix list statistics.

----End

10.10 Configuration Examples

The configuration examples in this section explain the networking requirements, networking diagram, configuration notes, configuration roadmap, and configuration procedure for different types of routing policies.

10.10.1 Example for Filtering Received and Advertised Routes

Filters can be applied to the received and advertised routes according to networking requirements.

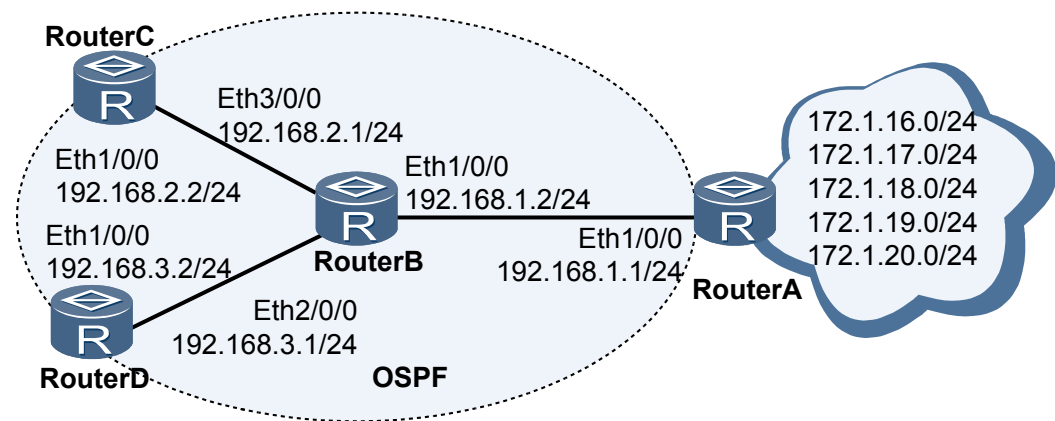
Networking Requirements

As shown in [Figure 10-1](#), in the network that runs OSPF, Router A receives routes from the network, and provides some of these routes for Router B. Router A is required to provide only 172.1.17.0/24, 172.1.18.0/24 and 172.1.19.0/24 for Router B. Router C is required to receive only 172.1.18.0/24. Router D receives all the routes provided by Router B.

 **NOTE**

AR150/200 is RouterC, or RouterD.

Figure 10-1 Networking diagram for filtering received and advertised routes



Configuration Roadmap

The configuration roadmap is as follows:

1. Configure basic OSPF functions on Router A, Router B, Router C, and Router D.
2. Configure static routes on Router A, and import these routes to OSPF.
3. Configure the policy for advertising routes on Router A, and check the filtering result on Router B.
4. Configure the policy for receiving routes on Router C, and check the filtering result on Router C.

Data Preparation

To complete the configuration, you need the following data:

- Five static routes imported by Router A.
- Router A, Router B, Router C, and Router D that reside in Area 0, that is the backbone area.
- Name of the IP prefix list and route to be filtered.

Procedure

Step 1 Assign an IP address to each interface.

The configuration details are not mentioned here.

Step 2 Configure basic OSPF functions.

Configure Router A.

```
[RouterA] ospf
[RouterA-ospf-1] area 0
[RouterA-ospf-1-area-0.0.0.0] network 192.168.1.0 0.0.0.255
[RouterA-ospf-1-area-0.0.0.0] quit
[RouterA-ospf-1] quit
```

Configure Router B.

```
[RouterB] ospf
[RouterB-ospf-1] area 0
```



```
[RouterB-ospf-1-area-0.0.0.0] network 192.168.1.0 0.0.0.255
[RouterB-ospf-1-area-0.0.0.0] network 192.168.2.0 0.0.0.255
[RouterB-ospf-1-area-0.0.0.0] network 192.168.3.0 0.0.0.255
[RouterB-ospf-1-area-0.0.0.0] quit
```

Configure Router C.

```
[RouterC] ospf
[RouterC-ospf-1] area 0
[RouterC-ospf-1-area-0.0.0.0] network 192.168.2.0 0.0.0.255
[RouterC-ospf-1-area-0.0.0.0] quit
[RouterC-ospf-1] quit
```

Configure Router D.

```
[RouterD] ospf
[RouterD-ospf-1] area 0
[RouterD-ospf-1-area-0.0.0.0] network 192.168.3.0 0.0.0.255
[RouterD-ospf-1-area-0.0.0.0] quit
```

Step 3 Configure five static routes on Router A and import these routes to OSPF.

```
[RouterA] ip route-static 172.1.16.0 24 NULL0
[RouterA] ip route-static 172.1.17.0 24 NULL0
[RouterA] ip route-static 172.1.18.0 24 NULL0
[RouterA] ip route-static 172.1.19.0 24 NULL0
[RouterA] ip route-static 172.1.20.0 24 NULL0
[RouterA] ospf
[RouterA-ospf-1] import-route static
[RouterA-ospf-1] quit
```

Check the IP routing table on Router B. You can view that the five static routes are imported to OSPF.

```
[RouterB] display ip routing-table
Route Flags: R - relay, D - download to fib
-----
Routing Tables: Public
  Destinations : 16          Routes : 16
Destination/Mask    Proto    Pre  Cost    Flags NextHop         Interface
127.0.0.0/8         Direct  0     0       D    127.0.0.1       InLoopBack0
127.0.0.1/32        Direct  0     0       D    127.0.0.1       InLoopBack0
172.1.16.0/24       O_ASE   150   1       D    192.168.1.1     Ethernet1/0/0
172.1.17.0/24       O_ASE   150   1       D    192.168.1.1     Ethernet1/0/0
172.1.18.0/24       O_ASE   150   1       D    192.168.1.1     Ethernet1/0/0
172.1.19.0/24       O_ASE   150   1       D    192.168.1.1     Ethernet1/0/0
172.1.20.0/24       O_ASE   150   1       D    192.168.1.1     Ethernet1/0/0
192.168.1.0/24      Direct  0     0       D    192.168.1.2     Ethernet1/0/0
192.168.1.1/32      Direct  0     0       D    192.168.1.1     Ethernet1/0/0
192.168.1.2/32      Direct  0     0       D    127.0.0.1       InLoopBack0
192.168.2.0/24      Direct  0     0       D    192.168.2.1     Ethernet3/0/0
192.168.2.1/32      Direct  0     0       D    127.0.0.1       InLoopBack0
192.168.2.2/32      Direct  0     0       D    192.168.2.2     Ethernet3/0/0
192.168.3.0/24      Direct  0     0       D    192.168.3.1     Ethernet2/0/0
192.168.3.1/32      Direct  0     0       D    127.0.0.1       InLoopBack0
192.168.3.2/32      Direct  0     0       D    192.168.3.2     Ethernet2/0/0
```

Step 4 Configure the policy for advertising routes.

Configure the IP prefix list named **a2b** on Router A.

```
[RouterA] ip ip-prefix a2b index 10 permit 172.1.17.0 24
[RouterA] ip ip-prefix a2b index 20 permit 172.1.18.0 24
[RouterA] ip ip-prefix a2b index 30 permit 172.1.19.0 24
```

Configure the policy for advertising routes on Router A and use the IP prefix list named **a2b** to filter routes.

```
[RouterA] ospf
[RouterA-ospf-1] filter-policy ip-prefix a2b export static
```

Check IP routing table on Router B, and you can view the three routes received by Router B from **a2b**.

```
[RouterB] display ip routing-table
Route Flags: R - relay, D - download to fib
-----
Routing Tables: Public
  Destinations : 14          Routes : 14
Destination/Mask    Proto Pre  Cost   Flags NextHop         Interface
 127.0.0.0/8        Direct 0    0      D    127.0.0.1          InLoopBack0
 127.0.0.1/32       Direct 0    0      D    127.0.0.1          InLoopBack0
 172.1.17.0/24     O_ASE 150  1      D    192.168.1.1       Ethernet1/0/0
 172.1.18.0/24     O_ASE 150  1      D    192.168.1.1       Ethernet1/0/0
 172.1.19.0/24     O_ASE 150  1      D    192.168.1.1       Ethernet1/0/0
 192.168.1.0/24     Direct 0    0      D    192.168.1.2        Ethernet1/0/0
 192.168.1.1/32     Direct 0    0      D    192.168.1.1        Ethernet1/0/0
 192.168.1.2/32     Direct 0    0      D    127.0.0.1          InLoopBack0
 192.168.2.0/24     Direct 0    0      D    192.168.2.1        Ethernet3/0/0
 192.168.2.1/32     Direct 0    0      D    127.0.0.1          InLoopBack0
 192.168.2.2/32     Direct 0    0      D    192.168.2.2        Ethernet3/0/0
 192.168.3.0/24     Direct 0    0      D    192.168.3.1        Ethernet2/0/0
 192.168.3.1/32     Direct 0    0      D    127.0.0.1          InLoopBack0
 192.168.3.2/32     Direct 0    0      D    192.168.3.2        Ethernet2/0/0
```

Step 5 Configure the policy for receiving routes.

Configure the IP prefix list named **in** on Router C.

```
[RouterC] ip ip-prefix in index 10 permit 172.1.18.0 24
```

Configure the policy for receiving routes on Router C, and use IP prefix list named **in** to filter routes.

```
[RouterC] ospf
[RouterC-ospf-1] filter-policy ip-prefix in import
```

Check the IP routing table on Router C, and you can find that Router C in the local core routing table receives only one route from the IP prefix list named **in**.

```
[RouterC] display ip routing-table
Route Flags: R - relay, D - download to fib
-----
Routing Tables: Public
  Destinations : 6          Routes : 6
Destination/Mask    Proto Pre  Cost   Flags NextHop         Interface
 127.0.0.0/8        Direct 0    0      D    127.0.0.1          InLoopBack0
 127.0.0.1/32       Direct 0    0      D    127.0.0.1          InLoopBack0
 172.1.18.0/24     O_ASE 150  1      D    192.168.2.1       Ethernet1/0/0
 192.168.2.0/24     Direct 0    0      D    192.168.2.2        Ethernet1/0/0
 192.168.2.1/32     Direct 0    0      D    192.168.2.1        Ethernet1/0/0
 192.168.2.2/32     Direct 0    0      D    127.0.0.1          InLoopBack0
```

Check the IP routing table on Router D, and you can find that Router D in the local core routing table receives all the routes advertised by Router B.

```
[RouterD] display ip routing-table
Route Flags: R - relay, D - download to fib
-----
Routing Tables: Public
  Destinations : 10         Routes : 10
Destination/Mask    Proto Pre  Cost   Flags NextHop         Interface
 127.0.0.0/8        Direct 0    0      D    127.0.0.1          InLoopBack0
 127.0.0.1/32       Direct 0    0      D    127.0.0.1          InLoopBack0
 172.1.17.0/24      O_ASE 150  1      D    192.168.3.1        Ethernet1/0/0
 172.1.18.0/24      O_ASE 150  1      D    192.168.3.1        Ethernet1/0/0
 172.1.19.0/24      O_ASE 150  1      D    192.168.3.1        Ethernet1/0/0
 192.168.1.0/24     OSPF  10   1      D    192.168.3.1        Ethernet1/0/0
```

```
192.168.2.0/24 OSPF 10 1 D 192.168.3.1 Ethernet1/0/0
192.168.3.0/24 Direct 0 0 D 192.168.3.2 Ethernet1/0/0
192.168.3.1/32 Direct 0 0 D 192.168.3.1 Ethernet1/0/0
192.168.3.2/32 Direct 0 0 D 127.0.0.1 Ethernet1/0/0
```

Check the OSPF routing table of Router C. You can find that three routes defined by the IP prefix list named **a2b** are in the OSPF routing table. In the link state protocol, you can run the **filter-policy import** command to filter the routes that join the local core routing table from the protocol routing table.

```
[RouterC] display ospf routing
```

```
OSPF Process 1 with Router ID 192.168.2.2
Routing Tables
```

```
Routing for Network
```

Destination	Cost	Type	NextHop	AdvRouter	Area
192.168.2.0/24	1	Stub	192.168.2.2	192.168.2.2	0.0.0.0
192.168.1.0/24	2	Stub	192.168.2.1	192.168.2.1	0.0.0.0
192.168.3.0/24	2	Stub	192.168.2.1	192.168.2.1	0.0.0.0

```
Routing for ASEs
```

Destination	Cost	Type	Tag	NextHop	AdvRouter
172.1.17.0/24	1	Type2	1	192.168.2.1	192.168.1.1
172.1.18.0/24	1	Type2	1	192.168.2.1	192.168.1.1
172.1.19.0/24	1	Type2	1	192.168.2.1	192.168.1.1

```
Total Nets: 6
```

```
Intra Area: 3 Inter Area: 0 ASE: 3 NSSA: 0
```

----End

Configuration Files

- Configuration file of Router A

```
#
sysname RouterA
#
interface Ethernet1/0/0
ip address 192.168.1.1 255.255.255.0
#
ospf 1
filter-policy ip-prefix a2b export static
import-route static
area 0.0.0.0
network 192.168.1.0 0.0.0.255
#
ip ip-prefix a2b index 10 permit 172.1.17.0 24
ip ip-prefix a2b index 20 permit 172.1.18.0 24
ip ip-prefix a2b index 30 permit 172.1.19.0 24
#
ip route-static 172.1.16.0 255.255.255.0 NULL0
ip route-static 172.1.17.0 255.255.255.0 NULL0
ip route-static 172.1.18.0 255.255.255.0 NULL0
ip route-static 172.1.19.0 255.255.255.0 NULL0
ip route-static 172.1.20.0 255.255.255.0 NULL0
#
return
```

- Configuration file of Router B

```
#
sysname RouterB
#
interface Ethernet1/0/0
ip address 192.168.1.2 255.255.255.0
#
interface Ethernet2/0/0
```

```
    ip address 192.168.3.1 255.255.255.0
#
interface Ethernet3/0/0
 ip address 192.168.2.1 255.255.255.0
#
ospf 1
 area 0.0.0.0
  network 192.168.1.0 0.0.0.255
  network 192.168.2.0 0.0.0.255
  network 192.168.3.0 0.0.0.255
#
return
```

- Configuration file of Router C

```
#
 sysname RouterC
#
interface Ethernet1/0/0
 ip address 192.168.2.2 255.255.255.0
#
ospf 1
 filter-policy ip-prefix in import
 area 0.0.0.0
  network 192.168.2.0 0.0.0.255
#
 ip ip-prefix in index 10 permit 172.1.18.0 24
#
return
```

- Configuration file of Router D

```
#
 sysname RouterD
#
interface Ethernet1/0/0
 ip address 192.168.3.2 255.255.255.0
#
ospf 1
 area 0.0.0.0
  network 192.168.3.0 0.0.0.255
#
return
```