Unicast Routing

QSR-1920, QSR-2920, QSR-3920

Оглавление

1. ROUTING BASICS

1.1. Overview

After a device receives a packet through an interface, the device selects a route according to the destination of the route and then forwards the packet to another interface. This process is called routing. In network devices, routes are stored in a route table database. The packets search the route table to determine the next hop and output interface according to the destination of the packets. Routes are categorized into three types according to their sources.

- Direct route: The route is generated based on the interface address. After a user configures the IP address of an interface, the device generates a direct route of the network segment according to the IP address and mask.
- Static route: The route is manually configured by the user.
- Dynamic route: The route is discovered through the dynamic route discovery protocol. Based on whether the dynamic routing protocol is used within an autonomous domain, two types of dynamic routing protocols are available: Interior Gateway Protocol (IGP) and Exterior Gateway Protocol (EGP). Here an autonomous domain refers to a network which has a unified management organization and unified routing policy. A routing protocol that is used within an autonomous domain is an IGP. Common IGPs include Routing Information Protocol (RIP) and Open Shortest Path First (OSPF). EGPs are usually used for routing among multiple autonomous domains. A common EGP is BGP.

Routing supports load balancing, that is, multiple routes to the same destination. In forwarding packets, a device transmits packets in load balancing mode according to the route table search result.

1.2. Routing Basic Function Configuration

Table 1-1 Routing basic function list

Configuration Tasks	
Configure load balancing for routing.	Configure the maximum number of load balancing entries.
	Configure the calculation method for the load balancing.
Configure the capacity of routes for Virtual Route Forwarding (VRF) routes.	Configure the capacity of VRF routes.

1.2.1. Configure Load Balancing for Routing

Configuration Condition

None

Configure the Maximum Number of Load Balancing Entries

If the costs of several paths to one destination are the same, the paths form load balancing. Configuring the maximum number of load balancing entries helps to improve the link utility rate and reduce the load of links.

Table 1-2 Configuring the maximum number of load balancing entries

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Configure the maximum number of load balancing entries.	**route path-limit** *max-number*	Optional. By default, the maximum number of load balancing entries for routing is 6.

Configure Non-equivalent Load Balancing

When the bandwidths of the two paths forming load balancing are inconsistent, it is configured as non-equivalent load balancing, which helps to improve the utilization of the link and reduce the burden of the link.

Table 1-3 Configure the non-equivalent load balancing

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Configure the IPv4 non-equivalent load balancing	**ip load-sharing mode unequal-cost**	Optional By default, do not enable the non-equivalent load balancing.
Configure the IPv6 non-equivalent load balancing	**ipv6 load-sharing mode unequal-cost**	Optional By default, do not enable the non-equivalent load balancing.

Configure Calculation Method for Load Balancing

The load balancing has the following three calculation methods:

- Calculation method based on the source and destination IP address: The source IP address and destination IP address are used to identify a flow. Packets on the same flow choose the same path without disorder. When the load on the flow is unbalanced, the load on the line is unbalanced.

- Calculation method based on the source IP address: Only the source IP address is used to identify a flow. Packets on the same flow choose the same path without disorder. When the flow load is unbalanced, load on the line may be unbalanced.
- Calculation method based on packets: Packets to the same destination IP address choose different paths, attempting to realize load balancing on paths. However, the packet disorder may occur.

Table 1-4 Configure the calculation method for the load balancing

Step	Command	Description
Enter the global configuration mode	**configure terminal**	-
Configure the load balancing for IP4 packets	**ip load-sharing** { **per-destination** | **per-packet** | **per-source** }	Optional By default, the calculation method based on the source IP address and destination IP address is used.
Configure the load balancing for IP6 packets	**ipv6 load-sharing** { **per-destination** | **per-packet** | **per-source** }	Optional By default, the calculation method based on the source IP address and destination IP address is used.

1.2.2. Configure Capacity of VRF Routes

Configuration Condition

None

Configure the Capacity of VRF Routes

To ensure normal use of devices and prevents a large number of routes from consuming too many resources, you can use the **routing-table limit** command to limit the capacity of routes for each Virtual Route Forwarding (VRF). When the capacity of routes reaches the threshold, an alarm is generated.

Table 1-5 Configure the capacity of VRF routes

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-

Step	Command	Description
Enter the VRF configuration mode.	**ip vrf** *vrf-name*	-
Configure the capacity of VRF routes.	**routing-table limit** *limit-value* { *threshold-value* | **syslog-alert** }	Optional. By default, when the routes reach 80% of the capacity, print the alarm information.

Note:

- This command cannot limit the capacity of routes for global VRF.
- If the number of routing entries exceeds the threshold, new routing information gets lost.
- For VRF-related configuration, refer to MPLS L3VPN configuration manual.

1.2.3. Routing Basics Monitoring and Maintaining

Table 1-6 Routing basics monitoring and maintaining

Command	Description
clear { **ip | ipv6 } route** [**vrf** *vrf-name*] { *ip-address mask* | **all** }	Clear the specified IPv4 route or IPv6 route in the route table.
show { **ip | ipv6 } route** [**vrf** *vrf-name*] [**bgp** | **connected** | **irmp** | **isis** | **ospf** | **rip** | **static** | **statistic** [**all**] | *ip-address* { *mask* | *mask-len* }]	Display IPv4 route or IPv6 route information.
show route-path limit	Display the maximum number of load balancing paths

1.3. Route Basis Typical Configuration Example

1.3.1. Configure Traffic to Perform Non-Load Balancing

Network Requirements

- All devices run the OSPF protocol for route interaction. A route 200.0.0.0/24 in the IP network is advertised into OSPF.
- Device1 learns the route 200.0.0.0/24 from device2 and device3, and modifies the bandwidth of Device1 interface gigabitethernet1 so that device1 preferably learns the route 200.0.0.0/24 from Device2.

- The traffic from PC on Device1 to 200.0.0.0/24 can be forwarded from Device2 and Device3 to IP network at the same time according to the bandwidth ratio.

Network Topology

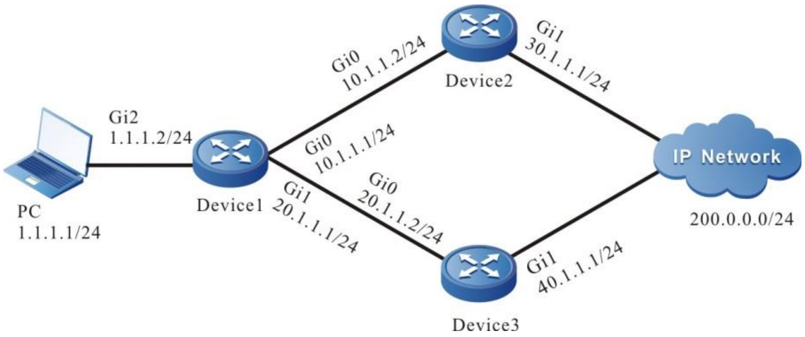

Figure 1-1 Networking of configuring traffic to perform the non-equivalent load balancing

Configuration Steps

Step 1: Configure the IP address of the interface (omitted).

Step 2: Configure OSPF.

#Configure Device1.

```
Device1#configure terminal
Device1(config)#router ospf 100
Device1(config-ospf)#router-id 1.1.1.1
Device1(config-ospf)#network 10.1.1.0 0.0.0.255 area 0
Device1(config-ospf)#network 20.1.1.0 0.0.0.255 area 0
Device1(config-ospf)#network 1.1.1.0 0.0.0.255 area 0
Device1(config-ospf)#exit
```

#Configure Device2.

```
Device2#configure terminal
Device2(config)#router ospf 100
Device2(config-ospf)#router-id 2.2.2.2
Device2(config-ospf)#network 10.1.1.0 0.0.0.255 area 0
Device2(config-ospf)#network 30.1.1.0 0.0.0.255 area 0
Device2(config-ospf)#exit
```

#Configure Device3.

```
Device3#configure terminal
Device3(config)#router ospf 100
Device3(config-ospf)#router-id 3.3.3.3
Device3(config-ospf)#network 20.1.1.0 0.0.0.255 area 0
Device3(config-ospf)#network 40.1.1.0 0.0.0.255 area 0
Device3(config-ospf)#exit
```

#View the OSPF neighbor information of Device1.

```
Device1#show ip ospf neighbor
OSPF process 1:
Neighbor ID    Pri   State        Dead Time   Address       Interface
2.2.2.2         1   Full/DR      00:00:35    10.1.1.2     gigabitethernet0
3.3.3.3         1   Full/DR      00:00:33    20.1.1.2     gigabitethernet1
```

#View the route table of Device1.

```
Device1#show ip route ospf
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
    U - Per-user Static route
    O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external
O   200.0.0.0/24 [110/2] via 10.1.1.2, 00:00:04, gigabitethernet0
             [110/2] via 20.1.1.2, 00:00:08, gigabitethernet1
```

There are two load balancing routes to network segment 200.0.0.0/24 in Device1 route table, and the forwarding paths to this network segment are Device1, Device2, Device1 and Device3 respectively.

Step 3: Modify the bandwidth of the interface.

#Modify the bandwidth of the connected interface between Device1 and Device3 to 200000 Kbps.

```
Device1(config)#interface gigabitethernet1
Device1(config-if-gigabitethernet1)#bandwidth 200000
Device1(config-if-gigabitethernet1)#exit
```

#Configure the non-equivalent load balancing.

```
Device1(config)#ip load-sharing mode unequal-cost
```

#Clear the OSPF process.

```
Device1#clear ip ospf process
Reset ALL OSPF process? [Yes|No]: Yes
 Clearing OSPF Process..............
 Done
```

Caution:

- After configuring non-equivalent load balancing, the process must be cleared before it can take effect.

Step 4: Check the result.

#Use the PC to send the changing traffic of 10000 sources and view the interface statistics.

```
Device1#show interface statistics
INTERFACE                              RATE(bits/sec)     BYTES          PACKETS
UTILIZATION
 --------------------------------------------------------------------------------
gigabitethernet0              INPUT   1175            3480          48            -
```

```
                          OUTPUT   168587        430689        3478         -
        gigabitethernet1    INPUT    97            242           3          -
                          OUTPUT   634453        810354        6536         -
```

It can be seen that the packets passed through interfaces gigabitethernet0 and gigabitethernet1 are 3478 and 6536 respectively, which is about equal to the bandwidth ratio of 1:2. The device realizes non-equivalent load balancing according to the bandwidth ratio.

2. STATIC ROUTES

2.1. Overview

A static route is a self-defined route which is manually configured by a user. It specifies a path for transmitting IP packets which are targeted at a specified destination.

Compared with dynamic routing, static routing has higher security and lower device resource occupancy. The disadvantage is that when the network topology changes, manual configuration is required, and there is no automatic re-configuration mechanism.

Static routes do not occupy line bandwidth or occupy CPU to calculate and advertise routes periodically, improving the device and network performance.

Static routes can be used to ensure the security of a small-scale network, for example, in a network where there is only one path connecting to an external network. In a large-scale network, static routes can implement security control on services or links of certain types. A majority of networks adopt dynamic routing protocols but you can still configure some static routes for special purposes.

Static routes can be re-distributed to a dynamic routing protocol, but dynamic routes cannot be re-distributed to static routes. Note that improper static route configuration may cause routing loops.

The default route is a special route which can be a static route. In a route table, the default route is a route to network 0.0.0.0 with the mask 0.0.0.0. You can use the **show ip route** command to check whether the route is valid. When the destination address of a received packet does not match any entry in the route table, the packet takes the default route. If no default route is available and the destination is not in the route table, the packet is discarded, and an ICMP packet is returned to the source end reporting that the destination address or network is not reachable. To prevent the route table from becoming too large, you can set a default route. The packet that fails to find a matching route table entry takes the default route for forwarding.

Null0 is a special route, with the route output interface as the Null0 interface. The Null0 interface is always in the UP status but it cannot forward packets. The packets that are sent to the interface will be discarded. If you configure a static route and specify the output interface for a certain network segment to Null0, the packets that are sent to the network segment will be discarded. Therefore, you can configure realize packet filtration by configuring Null0 static routes.

2.2. Static Routing Function Configuration

Table 2-1 Static route configuration function list

Configuration Tasks	
Configure a static route.	Configure a static route.
Configure the default administrative distance.	Configure the default administrative distance.
Configure load balancing routes.	Configure load balancing routes.

Configuration Tasks	
Configure a floating route.	Configure a floating route.
Configure a static route to coordinate with BFD.	Configure a static route to coordinate with BFD.
Configure a static route to coordinate with Track.	Configure a static route to coordinate with Track.
Configure the fast re-route of the static route	Configure the fast re-route of the static route
Configure the ISP operator route mapping	Configure the ISP operator route mapping

2.2.1. Configure Static Route

Configuration Condition

Before configuring a static route, ensure that:

- The link layer protocol has been configured to ensure normal communication at the link layer.
- Interface IP addresses have been configured so that neighbor nodes are reachable at the network layer.

Configure a Static Route

According to the parameters that have been specified, static routes are categorized into the following three types:

- Interface route: For an interface route, only the output interface is specified.
- Gateway route: For a gateway route, only the gateway address is specified.
- Interface gateway route: For an interface gateway route, both the output interface and the gateway address are specified.

Configured static routes become invalid if some of the following conditions are met:

The destination address is the local interface address.

The destination address is the network of the local direct interface.

The administrative distance of the route is 255.

The output interface of the route is DOWN.

No IP address has been configured for the output interface of the route.

The gateway address is not reachable.

The output interface and the gateway of the route conflict.

The output interface of the route does not exist.

The TRACK object that is associated with the route is "fake".

The status of the Bidirectional Forwarding Detection (BFD) session that is associated with the route is DOWN.

If an interface route meets any one condition among 1, 2, 3, 4, 5, 9, and 10, the route is invalid. If a gateway route meets any one condition among 1, 2, 3, 4, 6, 8, 9, and 10, the route is invalid. If an interface gateway route meets any of the above conditions, the route is invalid.

Table 2-2 Configuring an IPv4 static route

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Configure a static route.	**ip route** [**vrf** *vrf-name1*] *destination-ip-address destination-mask* { *interface-name* / [*nexthop-ip-address* [**vrf** *vrf-name2*]] } [**name** *nexthop-name*] [**tag** *tag-value*] [**track** *track-id*] [*administrative-distance*]	Mandatory The field *administrative-distance* is the administrative distance of the static route. If it is not specified, the default administrative distance is used.

Note:

- For a default route, the destination network and mask must be set to 0.0.0.0.
- The output interface of the Null0 route is Null0.
- The output interface of the Null0 interface need not be configured with an IP address.

2.2.2. Configure Default Administrative Distance

Configuration Condition

None

Configure Default Administrative Distance

The smaller the administrative distance that is specified for a static route in configuring the static route is, the higher the priority of the route is. If the administrative distance is not specified, the default administrative distance is used. You can modify the default administrative distance dynamically. After the default administrative distance is re-configured, the new default administrative distance is valid only for new static routes.

Table 2-3 Configuring the default administrative distance

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-

Step	Command	Description
Enters the static route configuration mode.	**router static**	-
Configure the default administrative distance.	**distance** *administration-distance*	Optional. The default value of the default administrative distance is 1.

Note:

- When you use the **ip route** command to configure a static route, you can specify an independent administrative distance for the route. If you do not specify the administrative distance, the default administrative distance is used.

2.2.3. Configure Recursion Function

Configuration Conditions

None

Configure Recursion Function

If the configured route gateway address must take effect through the gateway route reachable, you need to enable the recursion function of the static route to make the route take effect. By default, the recursive function of static route is enabled.

Table 2-4 Configure the recursion function

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enters the static route configuration mode.	**router static**	-
Configure the static route to support the recursion function	**recursion**	Optional By default, static route supports the route recursion function.

2.2.4. Configure Load Balancing Routes

Configuration Condition

None

Configure Load Balancing Routes

Load balancing routes means that multiple routes are configured to the same destination network. The output interfaces and the gateway addresses of the routes are different, but the administrative distances (priorities) of the routes are the same. Load balancing routes help to improve the link utility rate.

Table 2-5 Configuring the IPv4 load balancing routes

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Configure the first load balancing route.	**ip route** *destination-ip-address destination-mask interface-name1 distance*	Mandatory The output interface is interface-name1.
Configure the second load balancing route.	**ip route** *destination-ip-address destination-mask interface-name2 distance*	Mandatory. The output interface is interface-name2.

Note:

- In configuring load balancing routes, you must configure the values of distance for the routes to the same.

2.2.5. Configure Floating Route

Configuration Condition

None

Configure a Floating Route

Multiple routes are available to the same destination network. The output interfaces or gateway addresses of the routes are different, and the priorities of the routes are also different. The route with the higher priority becomes the primary route while the route with the lower priority becomes the floating route. In the route table, only the primary route is visible. The floating table appears in the route table only when the primary route becomes invalid. Therefore, the floating route is usually used as a backup route.

Table 2-6 Configuring an IPv4 floating route

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-

Step	Command	Description
Configure the primary route.	**ip route** *destination-ip-address destination-mask interface-name1 distance1*	Mandatory. The output interface of the primary route is *interface-name1* and the priority of the route is *distance1*.
Configure the floating route.	**ip route** *destination-ip-address destination-mask interface-name2 distance2*	Mandatory. The output interface of the floating route is *interface-name2*, the priority is *distance2*. The value of *distance2* must be larger than the value of *distance1*.

Note:

- In configuring the priorities of the routes, not that the smaller the *distance* value is, the higher the priority is.

2.2.6. Configure Static Route to Coordinate with BFD

Configuration Condition

None

Configure a Static Route to Coordinate with BFD

The Bidirectional Forwarding Detection (BFD) protocol provides a method for detecting the connectivity of the forwarding path between two adjacent routers with light load. A protocol neighbor can quickly detect the connectivity fault of a forwarding path. Different from other dynamic protocol routes, static routes cannot learn communication link failures. BFD provides a method for quickly detecting communication link failures for static routes. After a static route is configured to coordinate with BFD, fast switchover of routes can be implemented. Currently, a static route only supports the asynchronous BFD detection mode. Therefore, you need to configure the route to coordinate with BFD on the devices at the two end of the link.

If the status of BFD that is coordinated with the static route is DOWN, the static route becomes invalid.

Table 2-7 Configuring an IPv4 static route to coordinate with BFD

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-

Step	Command	Description
Configure a static route.	**ip route** *destination-ip-address destination-mask interface-name nexthop-ip-address*	Mandatory. Only the static route with both output interface and gateway address specified can coordinate with BFD.
Configure the output interface and the next-hop address for the route that is coordinated with BFD.	**ip route static bfd** *interface-name nexthop-ip-address*	Mandatory. The field *nexthop-ip-address* specifies the directly connected next-hop address.

Note:

- For introduction of BFD and how to configure its basic functions, refer to BFD configuration manual.

2.2.7. Configure Static Route to Coordinate with Track

Configuration Condition

None

Configure a Static Route to Coordinate with Track

Some modules in the system need to monitor some system information and then determine their working modes based on the information. The objects that are monitored by the other modules are called monitoring objects. To simplify the relations between the modules and monitoring objects, Track objects are used. A Track object can contain multiple monitoring objects, and it displays the comprehensive status of the monitoring object to external modules. The external modules are associated only with Track objects and they do not care about monitoring objects contained in the Track objects any more. A Track object has two statuses, "true" and "false". The external modules that are associated with the Track object determine its working modes according to the Track object status.

A static route can associate with a Track object to monitor system information and determine whether the route is valid according to the status reported by the Track object. If the Track object reports "true", the conditions required by the static route are satisfied, and the route is added to the route table. If the Track object reports "false", the route is deleted from the route table.

Table 2-8 Configuring an IPv4 static route to coordinate with track

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-

Step	Command	Description
Create a Track object and enter the configuration mode of the Track object.	**track** *track-id*	Mandatory
Configure the track object to monitor the link status of the specified interface.	**interface** *interface-name* **line-protocol**	Optional
Return to the global configuration mode.	**Exit**	-
Configure a static route and associate it with the Track object.	**ip route** *destination-ip-address destination-mask interface-name* **track** *track-id*	Mandatory When the link layer of the monitoring interface is UP, the route is valid; otherwise, the route is invalid.

2.2.8. Configure Fast Re-routing of Static Route

Configuration Condition

Before configuring the fast re-routing of the static route, first complete the following task:

- Use the route map to specify the egress interface and next hop of the standby route

Configure Fast Re-routing of Static Route

In the network using the static route, the traffic interruption caused by the link or device fault will continue until the protocol detects the link fault and the floating route takes effect. Usually, the traffic interruption lasts several seconds. To reduce the taffic interruption time, you can configure the fast re-routing of the static route. Set the backup next hop for the matched route by applying the route map. Once the master link fails, the traffic over the master link switches to the standby link at once, realizing the fast switching.

Table 2-9 Configure the fast re-routing of the static route

Step	Command	Description
Enter the system configuration mode	**configure terminal**	-

Step	Command	Description
Configure the fast re-routing of the static route	**ip route static fast-reroute route-map** *map-name*	Mandatory By default, do not enable the fast re-routing.
Configure the auto fast re-routing of the static route	**ip route static pic**	Mandatory By default, do not enable the auto fast re-routing.

Note:

- Only the static route specified with the egress interface and gateway address can perform the fast re-routing.
- When the fast re-routing based on route-map **set fast-reroute backup-nexthop** is auto, the protocol performs the auto fast re-routing.
- When using pic mode, the protocol performs the auto fast re-routing.
- The modes of enabling the fast re-routing are mutually exclusive.

2.2.9. Configure ISP Operator Route Mapping

Configuration Conditions

None

Configure Operator ISP Route Mapping

Configure the route mapping to the operator ISP. The configuration is saved in the form of a file and supports multiple operators. Support updating by the file or manually.

Table 2-10 Configure the operator ISP route mapping function

Step	Command	Description
Enter the global configuration mode	**configure terminal**	-
Enable the ISP route mapping function	**ip isp-mapping enable**	Mandatory
Configure the next-hop function of the ISP route	**ip isp-mapping isp-id** *ispid nexthop nexthop-ip-address* [**vrf** *vrf-name*] [**description** *description*]	Mandatory

2.2.10. Static Route Monitoring and Maintaining

Table 2-11 Static route monitoring and maintaining

Command	Description
show ip route [**vrf** *vrf-name*] **static**	Display the IPv4 static routes in the route table.
show running-config ip route	Display the configuration information about IPv4 static routes.

2.3. Typical Configuration Example of Static Routes

2.3.1. Configure Static Routing Basic Functions

Network Requirement

- On Device1, Device2 and Device3, configure static routes so that PC1 and PC2 can communicate with each other.

Network Topology

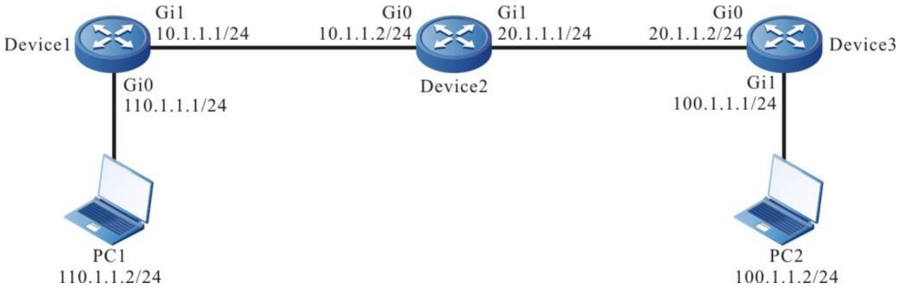

Figure 2-1 Networking for configuring static routing basic functions

Configuration Steps

Step 1: Configure the IP addresses of the interfaces. (Omitted)

Step 2: Configure static routes.

#Configure Device1.

```
Device1#configure terminal
Device1(config)#ip route 20.1.1.0 255.255.255.0 10.1.1.2
Device1(config)#ip route 100.1.1.0 255.255.255.0 10.1.1.2
```

#Configure Device2.

```
Device2#configure terminal
Device2(config)#ip route 110.1.1.0 255.255.255.0 10.1.1.1
Device2(config)#ip route 100.1.1.0 255.255.255.0 20.1.1.2
```

#Configure Device3.

```
Device3#configure terminal
```

```
Device3(config)#ip route 0.0.0.0 0.0.0.0 20.1.1.1
```

#Query the route table of Device1.

```
Device1#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

C   10.1.1.0/24 is directly connected, 00:06:47, gigabitethernet1
L   10.1.1.1/32 is directly connected, 00:06:47, gigabitethernet1
S   20.1.1.0/24 [1/10] via 10.1.1.2, 00:00:13, gigabitethernet1
S   100.1.1.0/24 [1/10] via 10.1.1.2, 00:00:05, gigabitethernet1
C   110.1.1.0/24 is directly connected, 00:08:21, gigabitethernet0
L   110.1.1.1/32 is directly connected, 00:08:21, gigabitethernet0
C   127.0.0.0/8 is directly connected, 28:48:33, lo0
L   127.0.0.1/32 is directly connected, 28:48:33, lo0
```

#Query the route table of Device2.

```
Device2#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

C   10.1.1.0/24 is directly connected, 00:00:37, gigabitethernet0
L   10.1.1.2/32 is directly connected, 00:00:37, gigabitethernet0
C   20.1.1.0/24 is directly connected, 00:00:27, gigabitethernet1
L   20.1.1.1/32 is directly connected, 00:00:27, gigabitethernet1
S   100.1.1.0/24 [1/10] via 20.1.1.2, 00:00:05, gigabitethernet1
S   110.1.1.0/24 [1/10] via 10.1.1.1, 00:00:13, gigabitethernet0
C   127.0.0.0/8 is directly connected, 30:13:18, lo0
L   127.0.0.1/32 is directly connected, 30:13:18, lo0
```

#Query the route table of Device3.

```
Device3#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

S   0.0.0.0/0 [1/10] via 20.1.1.1, 00:00:07, gigabitethernet0
C   20.1.1.0/24 is directly connected, 00:00:08, gigabitethernet0
```

```
L   20.1.1.2/32 is directly connected, 00:00:08, gigabitethernet0
C   100.1.1.0/24 is directly connected, 00:00:13, gigabitethernet1
L   100.1.1.1/32 is directly connected, 00:00:13, gigabitethernet1
C   127.0.0.0/8 is directly connected, 29:17:19, lo0
L   127.0.0.1/32 is directly connected, 29:17:19, lo0
```

Step 3: Check the result. Use the **ping** command to verify the connectivity between PC1 and PC2

#On PC1, use the ping command to check the connectivity.

```
C:\Documents and Settings\Administrator>ping 100.1.1.2

Pinging 100.1.1.2 with 32 bytes of data:

Reply from 100.1.1.2: bytes=32 time<1ms TTL=125
Reply from 100.1.1.2: bytes=32 time<1ms TTL=125
Reply from 100.1.1.2: bytes=32 time<1ms TTL=125
Reply from 100.1.1.2: bytes=32 time<1ms TTL=125

Ping statistics for 100.1.1.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

PC1 and PC2 can communicate with each other.

2.3.2. Configure Floating Static Route

Network Requirements

- On Device1, configure two static routes to reach network segment 192.168.1.0/24. One route passes Device2, and the other route passes Device3.
- Device1 first uses the route between Device1 and Device2 to forward packets. If the link is faulty, Device1 switches over to the route between Device1 and Device3 for communication.

Network Topology

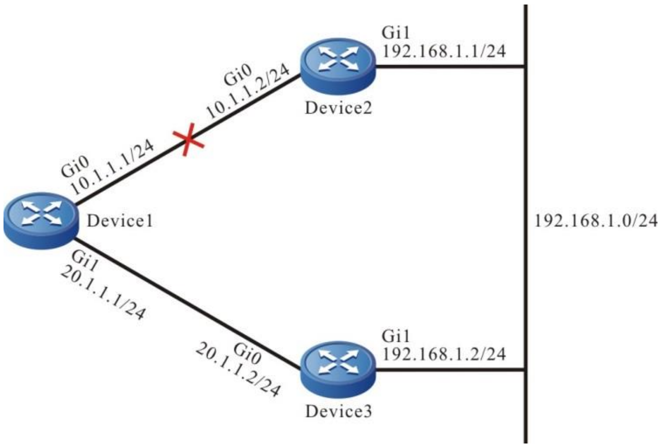

Figure 2-2 Networking for configuring a floating static route

Configuration Steps

Step 1: Configure the IP addresses of the interfaces. (Omitted)

Step 2: Configure static routes.

#On Device1, configure two routes to network segment 192.168.1.0/24 through Device2 and Device3.

```
Device1#configure terminal
Device1(config)#ip route 192.168.1.0 255.255.255.0 10.1.1.2
Device1(config)#ip route 192.168.1.0 255.255.255.0 20.1.1.2
```

#Query the route table of Device1.

```
Device1#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
    U - Per-user Static route
    O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external


C   10.1.1.0/24 is directly connected, 02:16:43, gigabitethernet0
L   10.1.1.1/32 is directly connected, 02:16:43, gigabitethernet0
C   20.1.1.0/24 is directly connected, 03:04:15, gigabitethernet1
L   20.1.1.1/32 is directly connected, 03:04:15, gigabitethernet1
C   127.0.0.0/8 is directly connected, 14:53:00, lo0
L   127.0.0.1/32 is directly connected, 14:53:00, lo0
S   192.168.1.0/24 [1/10] via 10.1.1.2, 00:00:05, gigabitethernet0
                   [1/10] via 20.1.1.2, 00:00:02, gigabitethernet1
```

According to the route tables, two routes from Device1 to network segment 192.168.1.0/24 are reachable, and the route form load balancing.

Step 3: Configure a floating route.

#Configure Device1. Modify the administrative range of the route with the gateway address 20.1.1.2 to 15 so that the route becomes a floating route.

```
Device1(config)#ip route 192.168.1.0 255.255.255.0 20.1.1.2 15
```

Step 4: Check the result.

#Query the route table of Device1.

```
Device1#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

C  10.1.1.0/24 is directly connected, 02:28:25, gigabitethernet0
L  10.1.1.1/32 is directly connected, 02:28:25, gigabitethernet0
C  20.1.1.0/24 is directly connected, 03:15:58, gigabitethernet1
L  20.1.1.1/32 is directly connected, 03:15:58, gigabitethernet1
C  127.0.0.0/8 is directly connected, 15:04:42, lo0
L  127.0.0.1/32 is directly connected, 15:04:42, lo0
S  192.168.1.0/24 [1/10] via 10.1.1.2, 00:11:47, gigabitethernet0
```

According to the route table, because the route with the administrative range 1 has a higher priority than the route with the administrative range 15, the route with the gateway 20.1.1.2 is deleted.

#After the route between Device1 and Device2 becomes faulty, query the route table of Device1.

```
Device1#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

C  20.1.1.0/24 is directly connected, 03:23:44, gigabitethernet1
L  20.1.1.2/32 is directly connected, 03:23:44, gigabitethernet1
C  127.0.0.0/8 is directly connected, 15:12:28, lo0
L  127.0.0.1/32 is directly connected, 15:12:28, lo0
S  192.168.1.0/24 [15/10] via 20.1.1.2, 00:00:02, gigabitethernet1
```

According to the route table, the route with a larger administrative range has been added to the route table to forward packets through Device3.

Note:

- The most significant feature of the floating static route is that it acts as a backup route.

2.3.3. Configure Static Route with the Null0 Interface

Network Requirements

- On Device1 and Device2, configure a default static route respectively, and configure the gateway addresses to the peer interface addresses of the two devices. On Device1, configure a static with the Null0 interface to filter only data that is sent to PC2.

Network Topology

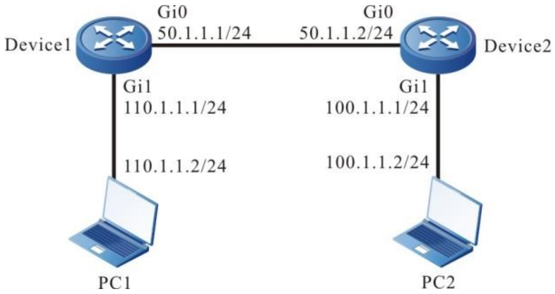

Figure 2-3 Networking for configuring a static route with the null0 interface

Configuration Steps

Step 1: Configure the IP addresses of the interfaces. (Omitted)

Step 2: Configure static routes.

#Configure Device1.

```
Device1#configure terminal
Device1(config)#ip route 0.0.0.0 0.0.0.0 50.1.1.2
```

#Configure Device2.

```
Device2#configure terminal
Device2(config)#ip route 0.0.0.0 0.0.0.0 50.1.1.1
```

#On PC1, use the ping command to check the connectivity with PC2.

```
C:\Documents and Settings\Administrator>ping 100.1.1.2

Pinging 100.1.1.2 with 32 bytes of data:

Reply from 100.1.1.2: bytes=32 time<1ms TTL=126
Reply from 100.1.1.2: bytes=32 time<1ms TTL=126
Reply from 100.1.1.2: bytes=32 time<1ms TTL=126
Reply from 100.1.1.2: bytes=32 time<1ms TTL=126

Ping statistics for 100.1.1.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

Step 3: Configure a static route with the Null0 interface.

#Configure Device1.

```
Device1(config)#ip route 100.1.1.2 255.255.255.255 null0
```

Step 4: Check the result.

#Query the route table of Device1.

```
Device1#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
    U - Per-user Static route
    O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

S   0.0.0.0/0 [1/10] via 50.1.1.2, 00:07:28, gigabitethernet0
C   50.1.1.0/24 is directly connected, 00:07:34, gigabitethernet0
L   50.1.1.1/32 is directly connected, 00:07:34, gigabitethernet0
C   110.1.1.0/24 is directly connected, 00:00:08, gigabitethernet1
L   110.1.1.1/32 is directly connected, 00:00:08, gigabitethernet1
C   127.0.0.0/8 is directly connected, 11:46:35, lo0
L   127.0.0.1/32 is directly connected, 11:46:35, lo0
S   100.1.1.2/32 [1/1] is directly connected, 00:02:31, null0
```

In the route table, the static route with the Null0 interface has been added.

#On PC1, use the **ping** command to check the connectivity with PC2.

```
C:\Documents and Settings\Administrator>ping 100.1.1.2

Pinging 100.1.1.2 with 32 bytes of data:


Request timed out.
Request timed out.
Request timed out.
Request timed out.


Ping statistics for 100.1.1.2:
  Packets: Sent = 4, Received = 0, Lost = 4 (100% loss)
```

The ICMP packets sent by PC1 are found to be targeted at interface Null0 according to the route table in Device1; therefore, the packets are discarded. In this way, PC1 fails to communicate with PC2.

Note:

- A static route with the Null0 interface is a special route. The packets sent to the Null0 interface will be discarded. Therefore, configuring a route with the Null0 interface implements packet filtration.

2.3.4. Configure Static Recursive Route

Network Requirements

- On Device1, configure two static routes to reach network segment 192.168.1.1/32. One route passes Device2, and the other passes Device3. Device1 first uses the route that passes Device3 to forward packets.
- On Device1, configure a static recursive route to reach network segment 200.0.0.0/24, with the gateway address being the loopback interface address 192.168.1.1 of Device3. After the route between Device1 and Device3 is faulty, Device1 switches to the route that passes Device2 for communication.

Network Topology

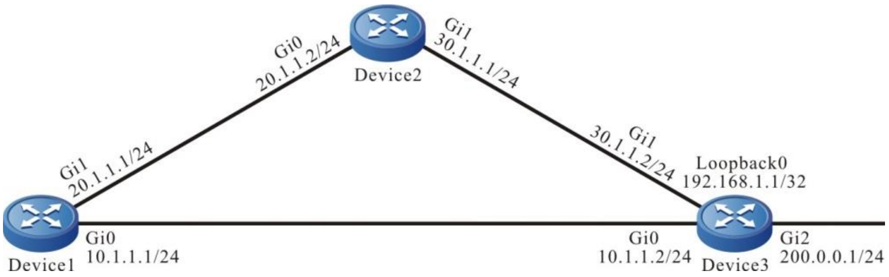

Figure 2-4 Networking for configuring a static recursive static route

Configuration Steps

Step 1: Configure the IP addresses of the interfaces. (Omitted)

Step 2: Configure static routes.

#Configure Device1.

```
Device1#configure terminal
Device1(config)#ip route 192.168.1.1 255.255.255.255 10.1.1.2
Device1(config)#ip route 192.168.1.1 255.255.255.255 20.1.1.2 10
```

#Configure Device2.

```
Device2#configure terminal
Device2(config)#ip route 192.168.1.1 255.255.255.255 30.1.1.2
```

Step 3: Configure a static recursive route.

#Configure Device1.

```
Device1(config)#ip route 200.0.0.0 255.255.255.0 192.168.1.1
```

#Query the route table of Device1.

```
Device1#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
```

```
          U - Per-user Static route
          O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

   C   10.1.1.0/24 is directly connected, 00:04:07, gigabitethernet0
   L   10.1.1.1/32 is directly connected, 00:04:07, gigabitethernet0
   C   20.1.1.0/24 is directly connected, 00:03:58, gigabitethernet1
   L   20.1.1.1/32 is directly connected, 00:03:58, gigabitethernet1
   C   127.0.0.0/8 is directly connected, 73:10:12, lo0
   L   127.0.0.1/32 is directly connected, 73:10:12, lo0
   S   200.0.0.0/24 [1/10] via 192.168.1.1, 00:00:08, gigabitethernet0
   S   192.168.1.1/32 [1/10] via 10.1.1.2, 00:01:46, gigabitethernet0
```

According to the route table, the gateway address of the route to 200.0.0.0/24 is 192.168.1.1, the output interface is gigabitethernet0, and the route relies on the route to 192.168.1.1/32.

Step 4: Check the result.

#After the route between Device1 and Device3 becomes faulty, query the route table of Device1.

```
   Device1#show ip route
   Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
          U - Per-user Static route
          O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

   C   20.1.1.0/24 is directly connected, 00:09:04, gigabitethernet1
   L   20.1.1.1/32 is directly connected, 00:09:04, gigabitethernet1
   C   127.0.0.0/8 is directly connected, 73:15:18, lo0
   L   127.0.0.1/32 is directly connected, 73:15:18, lo0
   S   200.0.0.0/24 [1/10] via 192.168.1.1, 00:00:02, gigabitethernet1
   S   192.168.1.1/32 [10/10] via 20.1.1.2, 00:00:02, gigabitethernet1
```

Comparing the route table information with the route table information in Step 3, the output interface of the route 200.0.0.0/24 has changed to gigabitethernet1, indicating that the route has been switched to the route to Device2.

2.3.5. Configure Static Route to Coordinate with BFD

Network Requirements

- On Device1, configure two static routes to network segment 201.0.0.0/24. One route passes Device2 while the other route passes Route3. Device first uses the route that passes Device 3 to forward packets. Similarly, on Device3, configure two static routes to network segment 200.0.0.0/24. Device3 first uses the route that passes Device1 to forward packets.
- On Device1 and Device3, configure static routes to coordinate with BFD. When the route between Device1 and Device3 is faulty, they can quickly switch over to the route that passes Device2.

Network Topology

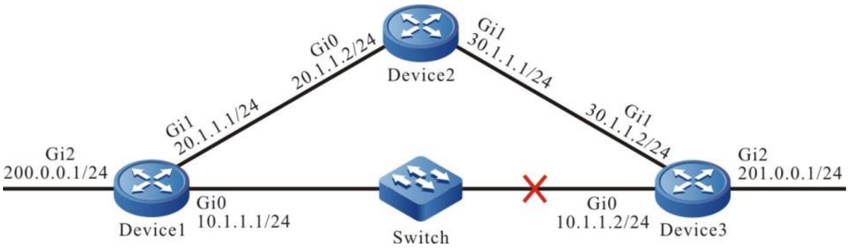

Figure 2-5 Networking for configuring a static route to coordinate with BFD

Configuration Steps

Step 1: Configure the IP addresses of the interfaces. (Omitted)

Step 2: Configure static routes.

#On Device1, configure two static routes to network segment 201.0.0.0/24.

```
Device1#configure terminal
Device1(config)#ip route 201.0.0.0 255.255.255.0 gigabitethernet0 10.1.1.2
Device1(config)#ip route 201.0.0.0 255.255.255.0 gigabitethernet1 20.1.1.2 10
```

#On Device2, configure two static routes to network segments 200.0.0.0/24 and 201.0.0.0/24 respectively.

```
Device2#configure terminal
Device2(config)#ip route 200.0.0.0 255.255.255.0 20.1.1.1
Device2(config)#ip route 201.0.0.0 255.255.255.0 30.1.1.2
```

#On Device3, configure two static routes to network segment 200.0.0.0/24.

```
Device3#configure terminal
Device3(config)#ip route 200.0.0.0 255.255.255.0 gigabitethernet0 10.1.1.1
Device3(config)#ip route 200.0.0.0 255.255.255.0 gigabitethernet1 30.1.1.1 10
```

#Query the route table of Device1.

```
Device1#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

C   10.1.1.0/24 is directly connected, 00:07:41, gigabitethernet0
L   10.1.1.1/32 is directly connected, 00:07:41, gigabitethernet0
C   20.1.1.0/24 is directly connected, 00:07:29, gigabitethernet1
L   20.1.1.1/32 is directly connected, 00:07:29, gigabitethernet1
C   127.0.0.0/8 is directly connected, 101:56:14, lo0
L   127.0.0.1/32 is directly connected, 101:56:14, lo0
C   200.0.0.0/24 is directly connected, 00:15:33, gigabitethernet2
```

```
L  200.0.0.1/32 is directly connected, 00:15:33, gigabitethernet2
S  201.0.0.0/24 [1/10] via 10.1.1.2, 00:02:23, gigabitethernet0
```

#Query the route table of Device3.

```
Device3#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

C  10.1.1.0/24 is directly connected, 00:10:21, gigabitethernet0
L  10.1.1.2/32 is directly connected, 00:10:21, gigabitethernet0
C  30.1.1.0/24 is directly connected, 00:10:09, gigabitethernet1
L  30.1.1.2/32 is directly connected, 00:10:09, gigabitethernet1
C  127.0.0.0/8 is directly connected, 126:44:08, lo0
L  127.0.0.1/32 is directly connected, 126:44:08, lo0
S  200.0.0.0/24 [1/10] via 10.1.1.1, 00:06:12, gigabitethernet0
C  201.0.0.0/24 is directly connected, 00:20:37, gigabitethernet2
L  201.0.0.1/32 is directly connected, 00:20:37, gigabitethernet2
```

Step 3: Configure a static route to coordinate with BFD.

#Configure Device1.

```
Device1(config)#ip route static bfd gigabitethernet0 10.1.1.2
```

#Configure Device3.

```
Device3(config)#ip route static bfd gigabitethernet0 10.1.1.1
```

Step 4: Check the result.

#Query the BFD session of Device1.

```
Device1#show bfd session
OurAddr              NeighAddr             LD/RD           State        Holddown
interface
10.1.1.1          10.1.1.2          15/22        UP        5000        gigabitethernet0
#Query the BFD session of Device3.
Device3#show bfd session
OurAddr               NeighAddr            LD/RD           State        Holddown
interface
10.1.1.2           10.1.1.1          22/15        UP        5000        gigabitethernet0
```

The BFD sessions have been normally set up on Device1 and Device3, indicating that the static routes are configured to coordinate with BFD successfully.

#When the route between Device1 and Device3 is faulty, BFD quickly detects the line fault and switch over to the route that passes Device2 for communication. Query the BFD session and route table of Device1.

```
Device1#show bfd session
OurAddr          NeighAddr          LD/RD          State          Holddown
interface
10.1.1.1      10.1.1.2      15/0      DOWN      5000      gigabitethernet0

Device1#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

C  10.1.1.0/24 is directly connected, 00:29:07, gigabitethernet0
L  10.1.1.1/32 is directly connected, 00:29:07, gigabitethernet0
C  20.1.1.0/24 is directly connected, 00:28:55, gigabitethernet1
L  20.1.1.1/32 is directly connected, 00:28:55, gigabitethernet1
C  127.0.0.0/8 is directly connected, 102:17:40, lo0
L  127.0.0.1/32 is directly connected, 102:17:40, lo0
C  200.0.0.0/24 is directly connected, 00:36:58, gigabitethernet2
L  200.0.0.1/32 is directly connected, 00:36:58, gigabitethernet2
S  201.0.0.0/24 [10/10] via 20.1.1.2, 00:00:09, gigabitethernet1
```

The BFD handling method on Device3 is the same as that on Device1.

2.3.6. Configure Fast Re-routing of Static Route

Network Requirements

- Device1 configures two static routes to the network 192.168.1.1/32. One is reachable via Device2 and the other is reachable via Device3. Device1 first uses the line with Device3 to forward the packet. Similarly, Device3 configures two static routes to the network 100.1.1.1/32. Device3 first uses the line with Device1 to forward the packet.
- Device1 and Device3 enable the fast re-routing of the static route. When the line between Device1 and Device3 fails, the service can switch to Device2 for communication fast.

Network Topology

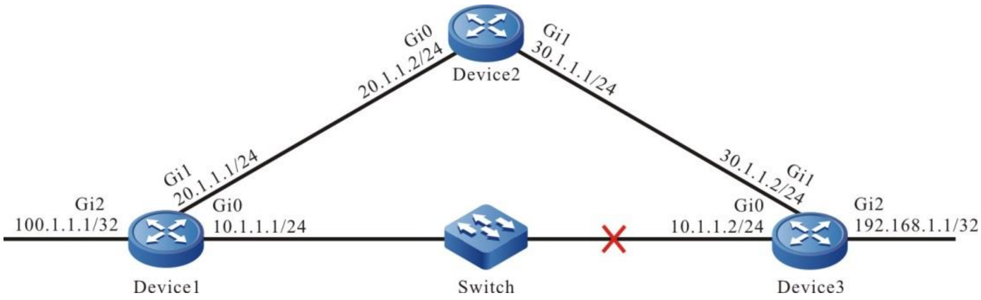

Figure 2-6 Configure the fast re-routing of the static route

Configuration Steps

Step 1: Configure the IP addresses of the interfaces. (Omitted)

Step 2: Configure static routes.

#Configure Device1: configure two static routes to the network 192.168.1.1/32. The management distance of the route with the gateway 20.1.1.2 is 10, making it become the floating route.

```
Device1#configure terminal
Device1(config)#ip route 192.168.1.1 255.255.255.255 gigabitethernet 0 10.1.1.2
Device1(config)#ip route 192.168.1.1 255.255.255.255 gigabitethernet 1 20.1.1.2 10
```

#Configure Device2: configure the static route to the network 100.1.1.1/32 and 192.168.1.1/32.

```
Device2#configure terminal
Device2(config)#ip route 192.168.1.1 255.255.255.255 30.1.1.2
Device2(config)#ip route 100.1.1.1 255.255.255.255 20.1.1.1
```

#Configure Device3: Configure two static routes to the network 100.1.1.1/32. The management distance of the route with the gateway 30.1.1.1 is 10, making it become the floating route.

```
Device3#configure terminal
Device3(config)#ip route 100.1.1.1 255.255.255.255 gigabitethernet 0 10.1.1.1
Device3(config)#ip route 100.1.1.1 255.255.255.255 gigabitethernet 1 30.1.1.1 10
```

Step 3: Configure the routing policy.

#Configure Device1: configure route-map to call the ACL only matching 192.168.1.1/32, while the other network is filtered. The route matching the match rule applies the backup next-hop interface gigabitethernet1 and the next-hop address is 20.1.1.2.

```
Device1(config)#ip access-list standard 1
Device1(config-std-nacl)#permit 192.168.1.1 0.0.0.0
Device1(config-std-nacl)#exit
Device1(config)#route-map ipfrr_static
Device1(config-route-map)#match ip address 1
Device1(config-route-map)#set fast-reroute backup-interface gigabitethernet1 backup-nexthop 20.1.1.2
```

```
Device1(config-route-map)#exit
```

#Configure Device3: configure route-map to call the ACL only matching 100.1.1.1/32, while the other network is filtered. The route matching the match rule applies the backup next-hop interface gigabitethernet1 and the next-hop address is 30.1.1.1.

```
Device3(config)#ip access-list standard 1
Device3(config-std-nacl)#permit 100.1.1.1 0.0.0.0
Device3(config-std-nacl)#exit
Device3(config)#route-map ipfrr_static
Device3(config-route-map)#match ip address 1
Device3(config-route-map)#set fast-reroute backup-interface gigabitethernet1 backup-nexthop 30.1.1.1
Device3(config-route-map)#exit
```

Step 4: Configure the fast re-routing.

#Configure Device1 to enable the fast re-routing of the static route.

```
Device1(config)#ip route static fast-reroute route-map ipfrr_static
```

#Configure Device3 to enable the fast re-routing of the static route.

```
Device3(config)#ip route static fast-reroute route-map ipfrr_static
```

Step 5: Check the result.

#View the route table of Device1.

```
Device1#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external
C   10.1.1.0/24 is directly connected, 00:31:17, gigabitethernet0
L   10.1.1.1/32 is directly connected, 00:31:17, gigabitethernet0
C   20.1.1.0/24 is directly connected, 00:08:43, gigabitethernet1
L   20.1.1.1/32 is directly connected, 00:08:43, gigabitethernet1
C   127.0.0.0/8 is directly connected, 24:43:25, lo0
L   127.0.0.1/32 is directly connected, 24:43:25, lo0
LC   100.1.1.1/32 is directly connected, 00:00:03, gigabitethernet2
S   192.168.1.1/32 [1/10] via 10.1.1.2, 00:19:12, gigabitethernet0
```

#View the fast re-route table of Device1. You can see the route of the network 192.168.1.1/32 and the next-hop interface is gigabitethernet1.

```
Device1#show ip frr route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
```

```
    U - Per-user Static route
    O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external
S   192.168.1.1/32 [1/10] via 20.1.1.2, 00:00:10, gigabitethernet1
```

#View the backup next-hop information of Device1 and you can see that the backup interface of the fast re-routing is gigabitethernet1.

```
Device1#show nexthop frr detail
Index          : 223
Type           : FRR
Reference Count     : 1
Active Path      : master
Nexthop Address     : 10.1.1.2
Interface       : gigabitethernet0
Interface Vrf     : global
Channel ID       : 10
Link Header Length   : 18
Link Header      : 00017a123453201201010101810000010800
Action         : FORWORDING
Slot          : 0
BK Nexthop Address   : 20.1.1.2
BK Interface      : gigabitethernet1
BK Interface Vrf    : global
BK Channel ID     : 11
BK Link Header Length  : 18
BK Link Header     : 00017a455449201201010102810000020800
BK Action       : FORWORDING
BK Slot        : 0


Total 1 entries.
```

#After the line between Device1 and Device3 fails, the system can fast detect and switch to Device2 for communication. View the route table and fast re-route table of Device1. The egress interface to the destination network 192.168.1.1/32 in the route table is switched to the backup interface gigabitethernet1.

```
Device1#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
    U - Per-user Static route
    O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external


C   10.1.1.0/24 is directly connected, 00:31:17, gigabitethernet0
```

```
L   10.1.1.1/32 is directly connected, 00:31:17, gigabitethernet0
C   20.1.1.0/24 is directly connected, 00:08:43, gigabitethernet1
L   20.1.1.1/32 is directly connected, 00:08:43, gigabitethernet1
C   127.0.0.0/8 is directly connected, 24:43:25, lo0
L   127.0.0.1/32 is directly connected, 24:43:25, lo0
LC   100.1.1.1/32 is directly connected, 00:00:03, gigabitethernet2
S   192.168.1.1/32 [10/10] via 20.1.1.2, 00:00:12, gigabitethernet1
```

The processing mode of Device3 is similar to Device1.

2.3.7. Configure ISP Operator Route Mapping

Network Requirements

- Device is connected to the operator's network, such as telecom operators. The export gateway is 192.168.202.254. After the operator's DNS server is configured and the ISP routing is enabled, the device can normally access the Internet.

Network Topology

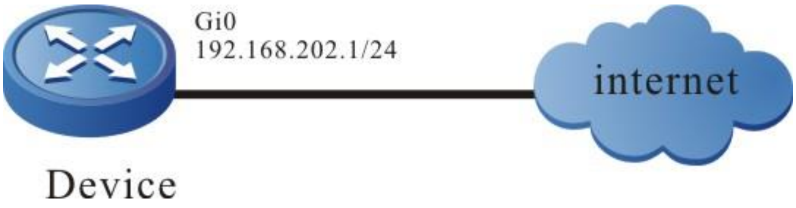

Figure 2-7 Networking of configuring the ISP operator route mapping

Configuration Steps:

Step 1: Configure the IP address of the interface (omitted).

Step 2: Configure the DNS server.

#On Device, configure the DNS server of the operator. For example, the DNS server address of Rostelecom is 212.48.193.36.

```
Device#configure terminal
Device(config)#ip name-server 212.48.193.36
```

#View the routing table on Device and ping www.mail.ru to check whether Internet can be connected.

```
Device#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

C   127.0.0.0/8 is directly connected, 2d:03:23:11, lo0
L   127.0.0.1/32 is directly connected, 2d:03:23:11, lo0
C   192.168.202.0/24 is directly connected, 2d:03:22:30, gigabitethernet0
L   192.168.202.1/32 is directly connected, 2d:03:22:30, gigabitethernet0
```

```
Device#
Device#ping www.mail.ru

%Bad IP address or unknown hostname!

Device#
```

It can be seen that there are only local routes on the router, no ISP operator routes, cannot resolve the domain name of www.mail.ru, and cannot access the Internet.

Step 3: Configure the ISP route.

#On Device, enable the ISP route, and configure the ISP route next hop to Telecom operator.

```
Device#configure terminal
Device(config)#ip isp-mapping enable
Device(config)#ip isp-mapping isp-id 2 next-hop 192.168.202.254
```

Note:

- The corresponding relationship between operator ID and operator is defined by itself.

Step 4: Check the result.

#On Device, view the route table, and ping www.mail.ru to check whether Internet can be connected.

```
Device#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

S  1.0.1.0/24 [1/0] via 192.168.202.254, 00:00:05, gigabitethernet0
S  1.0.2.0/23 [1/0] via 192.168.202.254, 00:00:05, gigabitethernet0
S  1.0.8.0/21 [1/0] via 192.168.202.254, 00:00:05, gigabitethernet0
S  1.0.32.0/19 [1/0] via 192.168.202.254, 00:00:05, gigabitethernet0
S  1.1.0.0/24 [1/0] via 192.168.202.254, 00:00:05, gigabitethernet0
S  1.1.2.0/23 [1/0] via 192.168.202.254, 00:00:05, gigabitethernet0
S  1.1.4.0/22 [1/0] via 192.168.202.254, 00:00:05, gigabitethernet0
S  1.1.9.0/24 [1/0] via 192.168.202.254, 00:00:05, gigabitethernet0
S  1.1.10.0/23 [1/0] via 192.168.202.254, 00:00:05, gigabitethernet0
S  1.1.12.0/22 [1/0] via 192.168.202.254, 00:00:05, gigabitethernet0
S  1.1.16.0/20 [1/0] via 192.168.202.254, 00:00:05, gigabitethernet0
S  1.1.32.0/19 [1/0] via 192.168.202.254, 00:00:05, gigabitethernet0
S  1.2.0.0/23 [1/0] via 192.168.202.254, 00:00:05, gigabitethernet0
```

```
S   1.2.5.0/24 [1/0] via 192.168.202.254, 00:00:05, gigabitethernet0
S   1.2.6.0/23 [1/0] via 192.168.202.254, 00:00:05, gigabitethernet0
S   1.2.9.0/24 [1/0] via 192.168.202.254, 00:00:05, gigabitethernet0
S   1.2.10.0/23 [1/0] via 192.168.202.254, 00:00:05, gigabitethernet0
S   1.2.12.0/22 [1/0] via 192.168.202.254, 00:00:05, gigabitethernet0
S   1.2.16.0/20 [1/0] via 192.168.202.254, 00:00:05, gigabitethernet0
```

You can see that ISP route is generated on Device.

#You can view that there is a route to the DNS server of Telecom operator on Device.

```
Device#show ip route 212.48.193.36
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

S   212.48.0.0/16 [1/0] via 192.168.202.254, 00:09:49, gigabitethernet0

Device#
#On Device, you can ping www.mail.ru.
Device#ping www.mail.ru

Press key (ctrl + shift + 6) interrupt it.
Sending 5, 76-byte ICMP Echos to 94.100.180.200 , timeout is 2 seconds:
!!!!!
Success rate is 100% (5/5). Round-trip min/avg/max = 16/22/46 ms.
```

3. IPV6 STATIC ROUTES

3.1. Overview

IPv6 static routing protocol and static routing protocol have the same behaviors except the IP address structure in the packet. Refer to the introduction of static routing.

3.2. IPv6 Static Routing Function Configuration

Table 3-1 IPv6 static route configuration function list

Configuration Tasks	
Configure an IPv6 static route.	Configure an IPv6 static route.
Configure IPv6 load balancing routes.	Configure IPv6 load balancing routes.
Configure an IPv6 floating route.	Configure an IPv6 floating route.
Configure an IPv6 static route to coordinate with Track.	Configure an IPv6 static route to coordinate with Track.
Configure fast re-routing of the IPv6 static route	Configure fast re-routing of the IPv6 static route

3.2.1. Configure an IPv6 Static Route

Configuration Condition

Before configuring an IPv6 static route, ensure that:

- The link layer protocol has been configured to ensure normal communication at the link layer.
- Interface IPv6 addresses have been configured so that neighbor nodes are reachable at the network layer.

Configure an IPv6 Static Route

According to the parameters that have been specified, IPv6 static routes are divided into the following three types:

- Interface route: For an interface route, only the output interface is specified.
- Gateway route: For a gateway route, only the gateway address is specified.
- Interface gateway route: For an interface gateway route, both the output interface and the gateway address are specified.

The configured IPv6 static routes become invalid if some of the following conditions are met:

1. The destination address is the local interface address.
2. The administrative distance of the route is 255.

3. The output interface of the route is DOWN.
4. The egress interface of the route does not enable IPv6.
5. The gateway address is not reachable.
6. The gateway address conflicts with the local address.
7. The output interface and the gateway of the route conflict.
8. The output interface of the route does not exist.
9. The TRACK object that is associated with the route is "fake".
10. The status of the BFDv6 session associated with the route is DOWN.

If an interface route meets any one condition among 1, 2, 3, 4, 8, 9, and 10, the route is invalid. If a gateway route meets any one condition among 1, 2, 5, 6, 9, and 10, the route is invalid. If an interface gateway route meets any of the above conditions, the route is invalid.

Table 3-2 Configuring an IPv6 static route

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Configure an IPv6 static route.	**ipv6 route** *destination-ipv6-address/destination-mask* { *interface-name* / [*nexthop-ipv6-address*] } [**name** *nexthop-name*] [**tag** *tag-value*] [**track** *track-id*] [*administrative-distance*]	Mandatory The field *administrative-distance* is the administrative distance of the static route. If it is not specified, the default administrative distance is used.

Note:

- When configuring the default route, the destination network and mask must be set to 0::/0.
- The output interface of the Null0 route should be configured to Null0.
- The output interface of the Null0 interface need not be configured with an IPv6 address.

3.2.2. Configure IPv6 Load Balancing Routes

Configuration Condition

None

Configure IPv6 Load Balancing Routes

IPv6 load balancing route means that there are multiple routes to the same destination network. The output interfaces and the gateway addresses of the routes are different, but the administrative distances (priorities) of the routes are the same. Load balancing routes help to improve the link utility rate.

Table 3-3 Configuring IPv6 load balancing routes

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Configure the IPv6 first load balancing route.	**ipv6 route** *destination-ipv6-address/destination-mask interface-name1 distance*	Mandatory The output interface is interface-name1.
Configure the IPv6 second load balancing route.	**ipv6 route** *destination-ipv6-address/destination-mask interface-name2 distance*	Mandatory. The output interface is interface-name2.

Note:

- In configuring load balancing routes, you must configure the values of *distance* for the routes to the same.

3.2.3. Configure IPv6 Floating Route

Configuration Condition

None

Configure an IPv6 Floating Route

IPv6 floating static route indicates there are multiple routes to the same destination network. The output interfaces or gateway addresses of the routes are different, and the priorities of the routes are also different. The route with the higher priority becomes the primary route while the route with the lower priority becomes the floating route. In the routing table, only the primary route is visible. The floating table appears in the routing table only when the primary route becomes invalid. Therefore, the floating route is usually used as a backup route.

Table 3-4 Configuring a floating route

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Configure the IPv6 primary route.	**ipv6 route** *destination-ipv6-address/destination-mask interface-name1 distance1*	Mandatory. The output interface of the primary route is *interface-name1* and the priority of the route is *distance1*.

Step	Command	Description
Configure the IPv6 floating route.	**ipv6 route** *destination-ipv6-address/destination-mask interface-name2 distance2*	Mandatory. The output interface of the floating route is *interface-name2*, the priority is *distance2*. The value of *distance2* must be larger than the value of *distance1*.

Note:

- In configuring the priorities of the routes, the smaller the *distance* value is, the higher the priority is.

3.2.4. Configure IPv6 Static Route to Coordinate with Track

Configuration Condition

None

Configure an IPv6 Static Route to Coordinate with Track

Some modules in the system need to monitor some system information and then determine their working modes based on the information. The objects that are monitored by the other modules are called monitoring objects. To simplify the relations between the modules and monitoring objects, Track objects are used. A Track object can contain multiple monitoring objects, and it displays the comprehensive status of the monitoring object to external modules. The external modules are associated only with Track objects and they do not care about monitoring objects contained in the Track objects any more. A Track object has two statuses, "true" and "false". The external modules that are associated with the Track object determine its working modes according to the Track object status.

A static route can associate with a Track object to monitor system information and determine whether the route is valid according to the status reported by the Track object. If the Track object reports "true", the conditions required by the static route are satisfied, and the route is added to the routing table. If the Track object reports "false", the route is deleted from the routing table.

Table 3-5 Configuring an IPv6 static route to coordinate with track

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Create a Track object and enter the configuration mode of the Track object.	**track** *track-id*	Mandatory

Step	Command	Description
Configure the track object to monitor the link status of the specified interface.	**interface** *interface-name* **line-protocol**	Optional By default, the track object is not configured to monitor the link status of the specified interface.
Return to the global configuration mode.	**exit**	-
Configure a static route and associate it with the Track object.	**Ipv6 route** *destination-ip-address destination-mask interface-name* **track** *track-id*	Mandatory When the link layer of the monitoring interface is UP, the route is valid; otherwise, the route is invalid.

3.2.5. Configure Fast Re-routing of IPv6 Static Route

Configuration Conditions

Before configuring the fast re-routing of the IPv6 static route, first complete the following task:

- When configuring the fast re-routing based on route-map, the associated route-map is already configured.

Configure Fast Re-routing of Static Route

In the network using IPv6 static routing, the traffic interruption caused by link or device failure will continue until the protocol detects the link failure and the floating route takes effect. The time often lasts for several seconds. In order to reduce the traffic interruption time, fast rerouting of the static route can be configured. By applying the route map, set the backup next hop for the successfully matched route. Once the active link fails, the traffic passing through the active link will be immediately switched to the standby link, so as to realize fast switching.

Table 3-6 Configure the fast re-routing of the IPv6 static route

Step	Command	Description
Enter the system configuration mode	**configure terminal**	-
Configure the fast re-routing of the static route based on route-map	**ipv6 route static fast-reroute route-map** *map-name*	Mandatory By default, do not enable the fast re-routing.

Step	Command	Description
Configure the auto fast re-routing of the static route	**ipv6 route static pic**	Mandatory’ By default, do not enable auto fast re-routing.

Note:

- Only the static routes specified with both out interface and gateway address can be rerouted quickly.
- Fast re-route based on route-map **set fast-reroute backup-nexthop** is auto, the protocol automatically reroutes fast.
- When the pic mode is used, the protocol automatically reroutes quickly.
- The various modes of enabling fast reroute are mutually exclusive.

3.2.6. IPv6 Static Route Monitoring and Maintaining

Table 3-7 IPv6 static route monitoring and maintaining

Command	Description
show ipv6 route [**vrf** *vrf-name*] **static**	Display the IPv6 static routes in the routing table.
show ipv6 static route [*ipv6-address*/*mask-length***]**	Display the IPv6 static route.
show running-config ipv6 route	Display the configuration information about IPv6 static routes.

3.3. Typical Configuration Examples of IPv6 Static Route

3.3.1. Configure Basic Functions of IPv6 Static Route

Network Requirements

- Device1, Device2, and Device3 configure the IPv6 static route, making PC1 and PC2 communicate with each other.

Network Topology

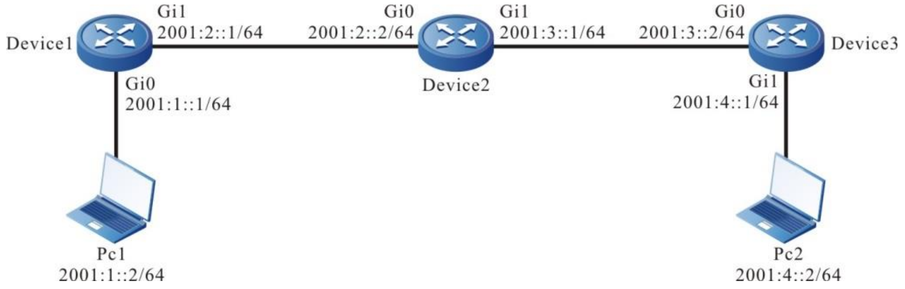

Figure 3-1 Networking for configuring the basic functions of the IPv6 static route

Configuration Steps

Step 1: Configure the IPv6 address of the interface (omitted).

Step 2: Configure the IPv6 static route.

#Configure the IPv6 route on Device1.

```
Device1#configure terminal
Device1(config)#ipv6 route 2001:4::/64 2001:2::2
```

#Configure Device2.

```
Device2#configure terminal
Device2(config)#ipv6 route 2001:1::/64 2001:2::1
Device2(config)#ipv6 route 2001:4::/64 2001:3::2
```

#On Device3, configure the IPv6 route.

```
Device3#configure terminal
Device3(config)#ipv6 route 2001:1::/64 2001:3::1
```

#Query the IPv6 route table of Device1.

```
Device1#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management

S   2001:4::/64 [1/10]
    via 2001:2::2, 00:03:14, gigabitethernet1
L   ::1/128 [0/0]
    via ::, 2w0d:01:09:06, lo0
C   2001:1::/64 [0/0]
    via ::, 00:25:55, gigabitethernet0
L   2001:1::1/128 [0/0]
    via ::, 00:25:53, lo0
C   2001:2::/64 [0/0]
    via ::, 04:01:46, gigabitethernet1
L   2001:2::1/128 [0/0]
    via ::, 04:01:45, lo0
```

#Query the IPv6 route table of Device2.

```
Device2#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management
```

```
L   ::1/128 [0/0]
    via ::, 5w2d:23:52:04, lo0
S   2001:1::/64 [1/10]
    via 2001:2::1, 00:02:56, gigabitethernet0
C   2001:2::/64 [0/0]
    via ::, 04:00:52, gigabitethernet0
L   2001:2::2/128 [0/0]
    via ::, 04:00:50, lo0
C   2001:3::/64 [0/0]
    via ::, 04:00:20, gigabitethernet1
L   2001:3::1/128 [0/0]
    via ::, 04:00:19, lo0
S   2001:4::/64 [1/10]
    via 2001:3::2, 00:02:36, gigabitethernet1
```

#Query the IPv6 route table of Device3.

```
Device3#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management

S   2001:1::/64 [1/10]
    via 2001:3::1, 00:00:08, gigabitethernet0
L   ::1/128 [0/0]
    via ::, 1w2d:20:54:36, lo0
C   2001:3::/64 [0/0]
    via ::, 03:58:28, gigabitethernet0
L   2001:3::2/128 [0/0]
    via ::, 03:58:26, lo0
C   2001:4::/64 [0/0]
    via ::, 00:11:13, gigabitethernet1
L   2001:4::1/128 [0/0]
    via ::, 00:11:12, lo0
```

Step 3: Check the result. Use the ping command to check the connectivity of PC1 and PC2.

#On PC1, use the ping command to check the connectivity.

```
C:\Documents and Settings\Administrator>ping 2001:4::2
Pinging 2001:4::2 with 32 bytes of data:
Reply from 2001:4::2: bytes=32 time<1ms TTL=128
Reply from 2001:4::2: bytes=32 time<1ms TTL=128
Reply from 2001:4::2: bytes=32 time<1ms TTL=128
Reply from 2001:4::2: bytes=32 time<1ms TTL=128
Ping statistics for 2001:4::2:
   Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
   Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

PC1 and PC2 can communicate with each other.

3.3.2. Configure IPv6 Static Floating Route

Network Requirements

- On Device1, configure two static routes to the segment 2001:3::/64: one is reachable via Device2, and the other is reachable via Device3.
- Device1 first uses the line with Device2 to forward the packet. When the line fails, switch to Device3 for communication.

Network Topology

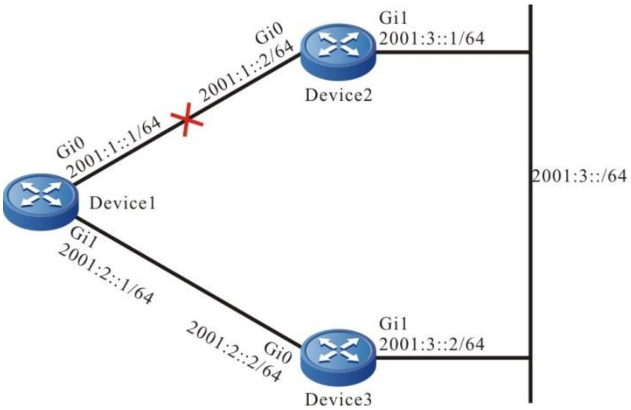

Figure 3-2 Networking for configuring the IPv6 static floating route

Configuration Steps

Step 1: Configure the IPv6 address of the interface (omitted).

Step 2: Configure the IPv6 static route.

#On Device1, configure two IPv6 static routes to the segment 2001:3::/64 passing Device2 and Device3 respectively.

```
Device1#configure terminal
Device1(config)#ipv6 route 2001:3::/64 2001:1::2
Device1(config)#ipv6 route 2001:3::/64 2001:2::2
```

#Query the IPv6 route table of Device1.

```
Device1#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
```

```
    O - OSPF, OE-OSPF External, M - Management


L  ::1/128 [0/0]
   via ::, 2w0d:02:13:16, lo0
C  2001:1::/64 [0/0]
   via ::, 00:22:33, gigabitethernet0
L  2001:1::1/128 [0/0]
   via ::, 00:22:32, lo0
C  2001:2::/64 [0/0]
   via ::, 00:17:47, gigabitethernet1
L  2001:2::1/128 [0/0]
   via ::, 00:17:46, lo0
S  2001:3::/64 [1/10]
   via 2001:1::2, 00:01:47, gigabitethernet0
          [1/10]
   via 2001:2::2, 00:01:36, gigabitethernet1
```

You can see that there are two routes to segment 2001:3::/64 on Device1, forming the load balance.

Step 3: Configure the IPv6 floating route.

#Configure Device1, modify the management distance of the route with the gateway 2001:2::2 as 15, making it become floating route.

```
Device1(config)#ipv6 route 2001:3::/64 2001:2::2 15
```

Step 4: Check the result.

#Query the IPv6 route table of Device1.

```
Device1#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
    U - Per-user Static route
    O - OSPF, OE-OSPF External, M - Management


L  ::1/128 [0/0]
   via ::, 2w0d:02:16:38, lo0
C  2001:1::/64 [0/0]
   via ::, 00:25:56, gigabitethernet0
L  2001:1::1/128 [0/0]
   via ::, 00:25:55, lo0
C  2001:2::/64 [0/0]
   via ::, 00:21:10, gigabitethernet1
```

```
L   2001:2::1/128 [0/0]
    via ::, 00:21:09, lo0
S   2001:3::/64 [1/10]
    via 2001:1::2, 00:05:10, gigabitethernet0
```

In the IPv6 route table, you can see that the route with the management distance 1 is prior to the route with the management distance 15, so the route with the gateway 2001:2::2 is deleted.

#After the line between Device1 and Device2 fails, query the route table of Device1.

```
Device1#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management

L   ::1/128 [0/0]
    via ::, 2w0d:02:21:06, lo0
C   2001:2::/64 [0/0]
    via ::, 00:25:38, gigabitethernet1
L   2001:2::1/128 [0/0]
    via ::, 00:25:37, lo0
S   2001:3::/64 [15/10]
    via 2001:2::2, 00:00:05, gigabitethernet1
```

In the IPv6 route table, you can see that the route with the larger management distance is added to the route table, and Device3 forwards the data.

Note:

- The largest feature of the static floating route is that it can back up the route.

3.3.3. Configure IPv6 Static NULL0 Interface Route

Network Requirements

- On Device1 and Device2, configure one static default route respectively, and the gateway addresses are the peer interface addresses of the two devices. On Device1, configure the static Null0 interface route, and can filter the data to PC2.

Network Topology

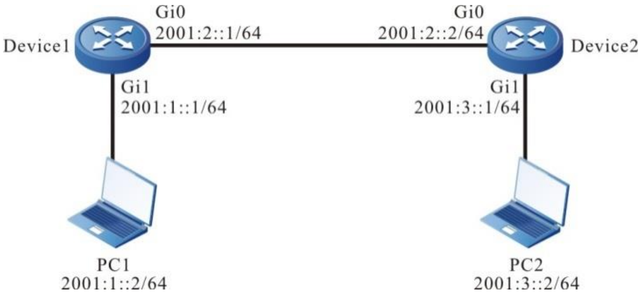

Figure 3-3 Networking for configuring IPv6 static NULL0 interface route

Configuration Steps

Step 1: Configure the IPv6 address of the interface (omitted).

Step 2: Configure the IPv6 static route.

#Configure Device1.

```
Device1#configure terminal
Device1(config)#ipv6 route ::/0 2001:2::2
```

#Configure Device2.

```
Device2#configure terminal
Device2(config)#ipv6 route ::/0 2001:2::1
```

#On PC1, use the ping command to check the connectivity.

```
C:\Documents and Settings\Administrator>ping 2001:3::2
Pinging 2001:3::2 with 32 bytes of data:
Reply from 2001:3::2: bytes=32 time<1ms TTL=128
Reply from 2001:3::2: bytes=32 time<1ms TTL=128
Reply from 2001:3::2: bytes=32 time<1ms TTL=128
Reply from 2001:3::2: bytes=32 time<1ms TTL=128
Ping statistics for 2001:3::2:
   Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
   Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

Step 3: Configure the IPv6 static Null0 interface route.

#Configure Device1.

```
Device1(config)#ipv6 route 2001:3::2/128 null0
```

Step 4: Check the result.

#Query the IPv6 route table of Device1.

```
Device1#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
      U - Per-user Static route
      O - OSPF, OE-OSPF External, M - Management

S   ::/0 [1/10]
    via 2001:1::2, 00:04:55, gigabitethernet1
L   ::1/128 [0/0]
    via ::, 2w0d:03:36:10, lo0
C   2001:1::/64 [0/0]
```

```
    via ::, 00:08:54, gigabitethernet1
L   2001:1::1/128 [0/0]
    via ::, 00:08:53, lo0
C   2001:2::/64 [0/0]
    via ::, 00:08:32, gigabitethernet0
L   2001:2::1/128 [0/0]
    via ::, 00:08:30, lo0
S   2001:3::2/128 [1/1]
    via ::, 00:00:34, null0
```

In the Ipv6 route table, the IPv6 static Null0 interface route is added.

#On PC1, use the ping command to check the connectivity with PC2.

```
C:\Documents and Settings\Administrator>ping 2001:3::2
Pinging 2001:3::2 with 32 bytes of data:
Request timed out.
Request timed out.
Request timed out.
Request timed out.
Ping statistics for 2001:3::2:
   Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
```

After searching for the route table for the ICMP packet sent by PC1 on Device1, discover that the egress interface is Null0, and directly drop. Therefore, PC1 cannot communicate with PC2.

Note:

- The static Null0 interface route is one special route, and the packets sent to the Null0 interface are all dropped. Therefore, configuring the static Null0 interface route can realize the filtering for the packets.

3.3.4. Configure IPv6 Static Recursive Route

Network Requirements

- On Device1, configure two static routes to the segment 192::3/128: one is reachable via Device2, and the other is reachable via Device3. Device1 first uses the line with Device3 to forward the packet.
- On Device1, configure one static recursive route to the segment 2001:4::/64, and the gateway address is the loopback interface address of Device3 192::3. After the line between Device1 and Device3 fails, the route can switch to Device2 for communication.

Network Topology

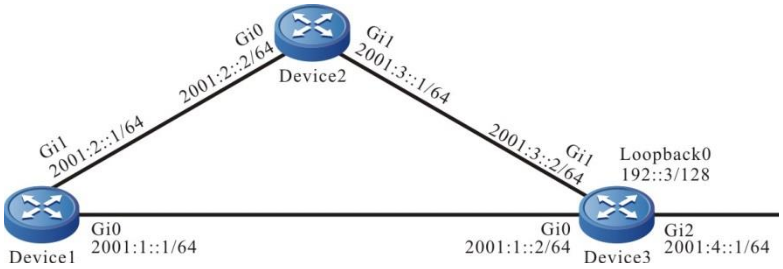

Figure 3-4 Networking for configuring IPv6 static recursive route

Configuration Steps

Step 1: Configure the IPv6 address of the interface (omitted).

Step 2: Configure the IPv6 static route.

#Configure Device1.

```
Device1#configure terminal
Device1(config)#ipv6 route 192::3/128 2001:1::2
Device1(config)#ipv6 route 192::3/128 2001:2::2 10
```

#Configure Device2.

```
Device2#configure terminal
Device2(config)#ipv6 route 192::3/128 2001:3::2
```

Step 3: Configure the IPv6 static recursive route.

# Configure Device1.

```
Device1(config)#ipv6 route 2001:4::/64 192::3
```

#Query the IPv6 route table of Device1.

```
Device1#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
     U - Per-user Static route
     O - OSPF, OE-OSPF External, M - Management

L   ::1/128 [0/0]
    via ::, 2w0d:03:12:46, lo0
S   192::3/128 [1/10]
    via 2001:1::2, 00:04:54, gigabitethernet0
C   2001:1::/64 [0/0]
    via ::, 00:22:47, gigabitethernet0
L   2001:1::1/128 [0/0]
    via ::, 00:22:45, lo0
C   2001:2::/64 [0/0]
```

```
     via ::, 00:16:16, gigabitethernet1
L    2001:2::1/128 [0/0]
     via ::, 00:16:15, lo0
S    2001:4::/64 [1/10]
     via 192::3, 00:00:43, gigabitethernet0
```

In the IPv6 route table, you can see that the gateway address of the route 2001:4::/64 is 192::3, the egress interface is gigabitethernet0, and the route depends on the route 192::3/128.

Step 4: Check the result.

#After the line between Device1 and Device3 fails, query the IPv6 route table of Device1.

```
Device1#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management

L    ::1/128 [0/0]
     via ::, 2w0d:03:17:48, lo0
S    192::3/128 [10/10]
     via 2001:2::2, 00:00:06, gigabitethernet1
C    2001:2::/64 [0/0]
     via ::, 00:21:18, gigabitethernet1
L    2001:2::1/128 [0/0]
     via ::, 00:21:17, lo0
S    2001:4::/64 [1/10]
     via 192::3, 00:00:06, gigabitethernet1
```

Compared with the route table of step 3, you can see that the egress interface of the route 2001:4::/64 is gigabitethernet1, indicating that the route already switches to Device2 for communication.

3.3.5. Configure Static Fast Re-routing of IPv6 Static Route

Network Requirements

- Device1 is configured with two static routes to the destination network segment 1001:2::1/64. One is reachable through Device2 and the other is reachable through Device3. Device1 preferentially uses the line with Device2 to forward packets.
- Static fast rerouting is enabled between Device1 and Device3. When the line fails, the service can be quickly switched to Device2 for communication.

Network Topology

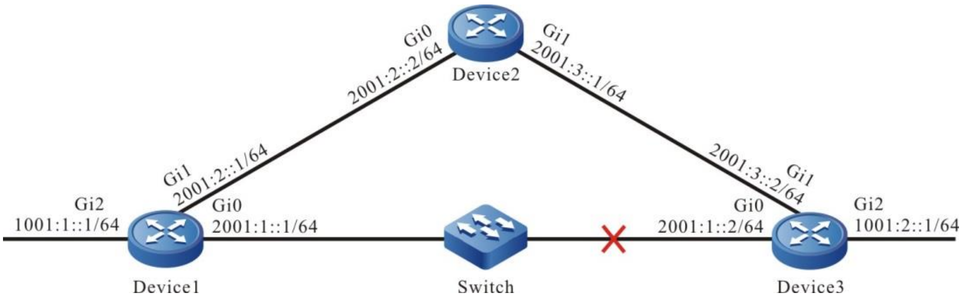

Figure 3-5 Networking of configuring the static fast re-routing of IPv6 static route

Configuration Steps

Step 1: Configure the IPv6 route of the interface (omitted).

Step 2: Configure the IPv6 static route.

#Configure Device1, and configure two statuc routes to the 1001:2::/64 network.

```
Device1#configure terminal
Device1(config)#ipv6 route 1001:2::/64 gigabitethernet 0 2001:1::2
Device1(config)#ipv6 route 1001:2::/64 gigabitethernet 1 2001:2::2 10
```

#Configure Device2, and configure one static route to the 1001:2::1/64 network.

```
Device2#configure terminal
Device2(config)#ipv6 route 1001:2::/64 gigabitethernet 1 2001:3::2
```

Step 3: Configure the echo function of ipv6 bfd on the gigabitethernet0 interface of Device3.

```
Device3#configure terminal
Device3(config)#interface gigabitethernet0
Device3(config-if-gigabitethernet0)# ipv6 bfd echo
Device3(config-if-gigabitethernet0)#exit
```

Step 4: Configure the routing policy.

#Configure Device1, configure route-map, and call ACL to match only 1001:2:: 1 / 64, while other network segments will be filtered out. The routing application matching the match rule backs up the next hop interface gigabitethernet1, and the next hop address 2001:2:: 2.

```
Device1(config)#ipv6 access-list extended 7001
Device1(config-v6-list)#permit ipv6 1001:2::1/64 any
Device1(config-v6-list)#exit
Device1(config)#route-map ipv6frr_st
Device1(config-route-map)#match ipv6 address 7001
Device1(config-route-map)#set ipv6 fast-reroute backup-interface gigabitethernet
1 backup-nexthop 2001:2::2
Device1(config-route-map)#exit
```

Step 5: Configure the static fast re-routing.

```
Device1(config)#ipv6 route static fast-reroute route-map ipv6frr_st
```

Step 6: Check the result.

#View the IPv6 static route table of Device1.

```
Device1#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management
L   ::1/128 [0/0]
    via ::, 04:26:25, lo0
C   1001:1::/64 [0/0]
    via ::, 04:21:32, gigabitethernet2
L   1001:1::1/128 [0/0]
    via ::, 04:21:32, gigabitethernet2
S   1001:2::/64 [1/10]
    via 2001:1::2, 00:03:00, gigabitethernet0
C   2001:1::/64 [0/0]
    via ::, 04:22:11, gigabitethernet0
L   2001:1::1/128 [0/0]
    via ::, 04:22:11, gigabitethernet0
C   2001:2::/64 [0/0]
    via ::, 04:20:50, gigabitethernet1
L   2001:2::1/128 [0/0]
    via ::, 04:20:50, gigabitethernet1
```

It can be seen from the routing table that route 1001:2:: / 64 preferably communicates with the line between Device1 and Device3.

#View the IPv6 frr route table of Device1.

```
Device1#show ipv6 frr route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management
S   1001:2::/64 [1/4294967295]
    via 2001:2::2, 00:04:32, gigabitethernet1
```

You can see that the next hop of the frr route 1001:2::/64 is 2001:2::2, and the out interface is gigabitethernet1.

#View the BFD session information of Device1.

```
Device1#show bfd session ipv6 detail
```

```
Total ipv6 session number: 1
OurAddr                    NeighAddr       LD/RD            State     Holddown
Interface
2001:1::1                    2001:1::2        1015/1015             UP          500
gigabitethernet0
Type:ipv6 direct  Mode:echo
Local Discriminator:67  Remote Discriminator:67
Local State:UP  Remote State:UP  Up for: 0h:30m:28s  Number of times UP:1
Send Interval:100ms  Detection time:500ms(100ms*5)
Local Diag:0  Demand mode:0  Poll bit:0
Registered protocols:FIB_MGR
Agent session info:
 Sender:slot 0  Recver:slot 0
```

You can see that FIB_MGR is linked with BFD successfully, the session is set up normally and the session mode is echo.

#After the line between Device1 and Device3 fails, it can quickly detect and switch to Device2 for communication. Check the routing table of Device1. In the routing table, the outgoing interface to the destination network 1001:2:: / 64 has been switched to the backup interface gigabitethernet1.

```
Device1#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management
L   ::1/128 [0/0]
    via ::, 04:56:34, lo0
C   1001:1::/64 [0/0]
    via ::, 04:51:41, gigabitethernet2
L   1001:1::1/128 [0/0]
    via ::, 04:51:41, gigabitethernet2
S   1001:2::/64 [10/10]
    via 2001:2::2, 00:00:08, gigabitethernet1
C   2001:2::/64 [0/0]
    via ::, 04:50:59, gigabitethernet1
L   2001:2::1/128 [0/0]
    via ::, 04:50:59, gigabitethernet1
```

3.3.6. Configure Dynamic Fast Re-routing of IPv6 Static Route

Network Requirements

- Device1 is configured with two static routes to the destination network segment 1001:2::1/64. One is reachable through Device2 and the other is reachable through Device3. Device1 preferentially uses the line with Device2 to forward packets.
- Dynamic fast rerouting is enabled between Device1 and Device3. When the line fails, the service can be quickly switched to Device2 for communication.

Network Topology

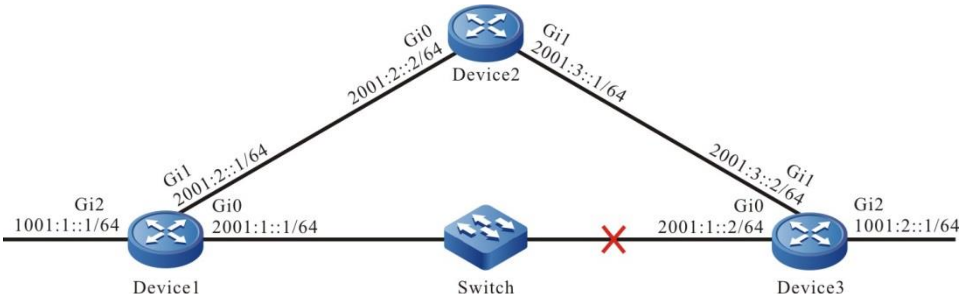

Figure 3-6 Networking of configuring the dynamic fast re-routing of IPv6 static route

Configuration Steps

Step 1: Configure the IPv6 route of the interface (omitted).

Step 2: Configure the IPv6 static route.

#Configure Device1, and configure two statuc routes to the 1001:2::/64 network.

```
Device1#configure terminal
Device1(config)#ipv6 route 1001:2::/64 gigabitethernet 0 2001:1::2
Device1(config)#ipv6 route 1001:2::/64 gigabitethernet 1 2001:2::2 10
```

#Configure Device2, and configure one static route to the 1001:2::1/64 network.

```
Device2#configure terminal
Device2(config)#ipv6 route 1001:2::/64 gigabitethernet 1 2001:3::2
```

Step 3: On the interface gigabitethernet0 of Device3, configure the echo function of ipv6 bfd.

```
Device3(config)#interface gigabitethernet0
Device3(config-if-gigabitethernet0)#ipv6 bfd echo
Device3(config-if-gigabitethernet0)#exit
```

Step 4: Configure the routing policy.

#Configure Device1, configure route-map, and call ACL to match only 1001:2::1/64, while other network segments will be filtered out. The routing application matching the match rule backs up the next hop interface gigabitethernet1, and the next hop address is auto.

```
Device1(config)#ipv6 access-list extended 7001
Device1(config-v6-list)#permit ipv6 1001:2::1/64 any
Device1(config-v6-list)#exit
```

```
Device1(config)#route-map ipv6frr_st
Device1(config-route-map)#match ipv6 address 7001
Device1(config-route-map)#set ipv6 fast-reroute backup-nexthop auto
Device1(config-route-map)#exit
```

Step 5: Configure the dynamic fast re-routing.

```
Device1(config)#ipv6 route static fast-reroute route-map ipv6frr_st
```

Step 6: Check the result.

#View the IPv6 static route table of Device1.

```
Device1#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management
L   ::1/128 [0/0]
    via ::, 04:26:25, lo0
C   1001:1::/64 [0/0]
    via ::, 04:21:32, gigabitethernet2
L   1001:1::1/128 [0/0]
    via ::, 04:21:32, gigabitethernet2
S   1001:2::/64 [1/10]
    via 2001:1::2, 00:03:00, gigabitethernet0
C   2001:1::/64 [0/0]
    via ::, 04:22:11, gigabitethernet0
L   2001:1::1/128 [0/0]
    via ::, 04:22:11, gigabitethernet0
C   2001:2::/64 [0/0]
    via ::, 04:20:50, gigabitethernet1
L   2001:2::1/128 [0/0]
    via ::, 04:20:50, gigabitethernet1
```

It can be seen from the routing table that route 1001:2::/64 preferably communicates with the line between Device1 and Device3.

#View the IPv6 frr route table of Device1.

```
Device1#show ipv6 frr route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management
S   1001:2::/64 [1/4294967295]
    via 2001:2::2, 00:04:32, gigabitethernet1
```

You can see that the next hop of the frr route 1001:2::/64 is 2001:2::2, and the out interface is gigabitethernet1.

#View the BFD session information of Device1.

```
Device1#show bfd session ipv6 detail
Total ipv6 session number: 1
OurAddr                     NeighAddr      LD/RD            State     Holddown
Interface
2001:1::1                      2001:1::2        1015/1015            UP         500
gigabitethernet0
Type:ipv6 direct  Mode:echo
Local Discriminator:67  Remote Discriminator:67
Local State:UP  Remote State:UP  Up for: 0h:30m:28s  Number of times UP:1
Send Interval:100ms  Detection time:500ms(100ms*5)
Local Diag:0  Demand mode:0  Poll bit:0
Registered protocols:FIB_MGR
Agent session info:
 Sender:slot 0  Recver:slot 0
```

You can see that FIB_MGR is linked with BFD successfully, the session is set up normally, and the session mode is echo.

#After the line between Device1 and Device3 fails, it can quickly detect and switch to Device2 for communication. Check the routing table of Device1. In the routing table, the outgoing interface to the destination network 1001:2:: / 64 has been switched to the backup interface gigabitethernet1.

```
Device1#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management
L   ::1/128 [0/0]
    via ::, 04:56:34, lo0
C   1001:1::/64 [0/0]
    via ::, 04:51:41, gigabitethernet2
L   1001:1::1/128 [0/0]
    via ::, 04:51:41, gigabitethernet2
S   1001:2::/64 [10/10]
    via 2001:2::2, 00:00:08, gigabitethernet1
C   2001:2::/64 [0/0]
    via ::, 04:50:59, gigabitethernet1
L   2001:2::1/128 [0/0]
    via ::, 04:50:59, gigabitethernet1
```

4. RIP

4.1. Overview

On the current Internet, it is impossible to run only one gateway protocol. You can divide it into multiple Autonomous Systems (ASs) and each has its own routing technology. The internal routing protocols within an AS are Interior Gateway Protocols (IGPs). Routing Information Protocol (RIP) is one type of IGP. RIP adopts the Vector-Distance algorithm. RIP features simple and easy-to-use, so it is widely used in numerous small-sized networks.

RIP has two versions: RIPv1 and RIPv2. RIPv1 does not support classless routing, and RIPv2 supports classless routing. Usually, RIPv2 is used.

RIP is a simple protocol which provides simple configuration. However, the number of routes to be advertised by RIP is directly proportional to the number of routes in the route table. If the number of routes is large, a lot of device resources and network resources are consumed. In addition, RIP specifies that the maximum number of hops that a routing path that passes routers is 15, so RIP is applicable only to simple small- and medium-sized network. RIP is applicable for most campus networks and LANs with a simple structure and strong continuity. For a more complex environment, RIP is not recommended.

RIPv1 was introduced earlier in RFC1058, but it has many deficiencies. To improve the deficiencies of RIPv1, RFC1388 introduced RIPv2, which was then revised in RFC 1723 and RFC 2453.

4.2. RIP Function Configuration

Table 4-1 RIP function list

Configuration Tasks	
Configure basic functions of RIP.	Enables RIP globally.
	Enable RIP for VRF.
	Configure RIP versions.
Configure RIP route generation.	Configure RIP to advertise the default route.
	Configure RIP to re-distribute routes.
Configure RIP route control.	Configure the administrative distance of RIP.
	Configure an RIP route summary.
	Configure the RIP metric offset.

Configuration Tasks	
Configure RIP route control.	Configure RIP route filtration.
	Configure the metric of the RIP interface.
	Configure the routing flag for an RIP interface.
	Configure the maximum load balancing for RIP.
Configure RIP network authentication.	Configure RIP network authentication.
Configure RIP network optimization.	Configure RIP timers.
	Configure RIP split horizon and toxicity reverse of RIP.
	Configure source address check.
	Configure a static RIP neighbor.
	Configure a passive RIP interface.
	Configure RIP to trigger updates.
	Configure an RIP backup interface.
Configure RIP to coordinate with BFD.	Configure RIP to coordinate with BFD.

4.2.1. Configure Basic Functions of RIP

Configuration Condition

Before configuring the basic functions of RIP, ensure that:

- The link layer protocol has been configured to ensure normal communication at the link layer.
- The network layer addresses of the interfaces have been configured so that the adjacent network nodes are reachable at the network layer.

Enable RIP Globally

Before using RIP, make the following configurations:

- Create an RIP process.
- Configure RIP to cover a directly connected network or interface.

Table 4-2 Enabling RIP globally

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Create an RIP process and enter the RIP configuration mode.	**router rip**	Mandatory. By default, the RIP process is disabled.
Configure RIP to cover a specified network segment or interface.	**network** { *ip-address* | *interface-name* }	Mandatory. By default, RIP does not cover any directly connected network or interface.

Note:

- The covered network segment is categorized into classful addresses.
- The **network** *ip-address* command cannot cover the super network addresses. To cover super network addresses, use the **network** *interface-name* command.
- **Enable RIP for VRF**

To enable RIP to support VRF functions, make the following configurations:

- Configure a VRF and add an interface to the VRF.
- Enable the RIP function in the VRF address family.
- Configure RIP to cover a VRF directly connected network or interface.

Table 4-3 Enable RIP for VRF

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Create an RIP process and enter the RIP configuration mode.	**router rip**	Mandatory. By default, the RIP process is disabled.

Step	Command	Description
Enter the VRF address family configuration mode of the RIP protocol.	**address-family** { **ipv4 vrf** *vrf-name* }	Mandatory. By default, the VRF address family mode is disabled.
Configure RIP to cover a specified network segment or interface.	**network** { *ip-address* | *interface-name* }	Mandatory. By default, RIP does not cover any directly connected network or interface.

Note:

- To enable RIP in VRF mode, you must first create VRF-related configurations.

Configure RIP Versions

RIP has two versions, RIPv1 and RIPv2. They can be configured in three modes: global configuration mode, VRF configuration mode, and interface configuration mode.

- By default, RIPv1 is enabled in global configuration mode and VRF configuration mode, and it is not configured in interface configuration mode.
- The version configuration command in interface configuration mode is a higher priority than the version configuration command in global or VRF configuration mode.
- If the version configuration command is not configured, the command in VRF configuration mode of the interface to which the VRF belongs or the command global configuration mode is used.
- In interface configuration mode, the RIP transmit version and the RIP receive version can be configured independently.
- After versions are configured, RIP has strict packet transmitting and receiving processing: In the case of RIPv1, the interface transmits and receives only RIPv1 broadcast and unicast packets. In the case of RIPv2, the interface can transmit and receive RIPv2 unicast, multicast, and broadcast packets. In the case of RIPv1 compatible mode, the interface can transmit RIPv2 unicast and broadcast packets.

Table 4-4 Configure RIP versions

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Create an RIP process and enter the RIP configuration mode.	**router rip**	Mandatory. By default, the RIP process is disabled.

Step	Command	Description
Configure the global RIP version.	**version** { 1 | 2 }	Mandatory. By default, RIPv1 is enabled.
Enter the RIP VRF configuration mode.	**address-family** { **ipv4 vrf** *vrf-name* }	Mandatory. By default, the VRF address family mode is disabled.
Configure the RIP version in RIP VRF configuration mode.	**version** { 1 | 2 | compatible }	Mandatory. By default, RIPv1 is enabled.
Return to the RIP configuration mode.	**exit-address-family**	-
Return to the global configuration mode.	**exit**	-
Enter the interface configuration mode.	**interface** *interface_name*	-
Configure the RIP transmit version of the interface.	**ip rip send version** {{ 1 / 2 } | 1-compatible }	Optional. By default, the interface transmits packets based on the global RIP version.
Configure the RIP receive version of the interface.	**ip rip receive version** { 1 / 2 }	Optional. By default, the interface receives packets based on the global RIP version.

4.2.2. Configure RIP Route Generation

Configuration Condition

Before configuring RIP route generation, ensure that:

- Interface IP addresses have been configured so that neighbor nodes are reachable at the network layer.
- RIP is enabled.

Configure RIP to Advertise the Default Route

Through configuration, a device can send the default route on all RIP interfaces to set itself as the default gateway of other neighbor devices.

Table 4-5 Configure RIP to advertise the default route

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the RIP configuration mode.	**router rip**	Mandatory. By default, the RIP process is disabled.
Configure RIP to advertise the default route.	**default-information only | originate [metric** *metric-value***]**	Mandatory. When only is configured, default routes are published when there are local default routes. When origin is configured, the default route will be published even if there is no local default route. By default, RIP does not advertise the default route.

Note:

- If a default route (0.0.0.0/0) is learnt, the default route (0.0.0.0/0) advertised by the local device is replaced. When loops exists in a network, network flapping may be caused. In using this command, prevent other devices in the same routing domain from enabling the command at the same time.

Configure RIP to Redistribute Routes

By redistributing routes, you can introduce the routes generated by other protocols to RIP.

Table 4-6 Configure RIP to redistribute routes

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-

Step	**Command**	**Description**
Enter the RIP configuration mode.	**router rip**	Mandatory. By default, the RIP process is disabled.
Configure the default metric for the routes of other protocols introduced to RIP.	**default-metric** *metric-value*	Optional. By default, the default metric of the introduced routes of other protocols is 1.
Configure RIP to redistribute routes.	**redistribute** *protocol* [*protocol-id*] [**metric** *metric-value*] [**route-map** *route-map-name*] [**match** *route-sub-type*]	Mandatory. By default, route redistribфution is not configured.

Note:

- If the metric command option is specified during redistribution, the redistributed route adopts the metric.
- In configuring RIP to redistribute routes, the available *match* options for the applied routing policy include ip address, route type, and tag, and the available set options for the applied routing policy include interface, ip next-hop, route source, and metric.

4.2.3. Configure RIP Route Control

Configuration Condition

Before configuring RIP route control, ensure that:

- Interface IP addresses have been configured so that neighbor nodes are reachable at the network layer.
- RIP is enabled.

Configure the Administrative Distance of RIP

One device can run multiple routing protocols at the same time. The device selects the optimal route from the routes that are learnt from different protocols based on the administrative distances. The smaller the administrative distance is, the higher the priority is.

Table 4-7 Configure the administrative distance of RIP

Step	**Command**	**Description**
Enter the global configuration mode.	**configure terminal**	-

Step	Command	Description
Enter the RIP configuration mode.	**router rip**	-
Configure the administrative distance of RIP.	**distance** *distance-value*	Mandatory. By default, the administrative distance of RIP is 120.

Configure an RIP Route Summary

Through RIP route summary, a routing device summarizes subnet routes in a natural network segment to form a summary route. The summary route and the original subnet routes all exist in the RIP route table.

After RIP route summary is configured, the device advertises only the route summary. This greatly decreases the size of adjacent RIP route tables in a medium- and large-sized network and decreases the consumption of the network bandwidth by routing protocol packets.

A route summary takes the minimum value among metrics of all subnet routes as its metric.

RIPv1 supports automatic route summary mode, and RIPv2 supports the automatic route summary mode and the manual summary mode.

1. RIP auto route summary

Different from manual route summary, auto route summary enables RIP to automatically generate a natural mask route based on subnet routes in one natural network segment.

Table 4-8 Configure the auto route summary function

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the RIP configuration mode.	**router rip**	Mandatory. By default, the RIP process is disabled.
Configure the auto route summary function of RIPv2.	**auto-summary**	Mandatory. By default, the auto route summary function of RIPv2 is disabled, but the auto route summary function of RIPv1 is enabled.

Note:

- RIPv1 does not support the route summary command.
- The tag of a route summary is 0, and minimum metric of the routes is taken as the route summary metric. If the auto route summary is configured, auto route summary has the priority.
- Exercise caution in using the auto route summary function. Ensure that it is necessary to perform auto route summary; otherwise, routing loops may be caused.
- When the auto route summary function of RIPv2 is enabled, if the interface of the advertised route and the route are in the same natural network segment, the update packet sent from the interface does not result in summary of all subnet routes in the natural network segment; otherwise, routes are gathered to form a natural network segment and then it is advertised.

1. Manual route summary

In manual route summary, a combination of a destination address and a mask need to be configured. The combination gathers all routes in the covered network segment for route summary.

Table 4-9 Configure the manual route summary function

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the interface configuration mode.	**interface** *interface-name*	-
Configure the manual route summary function of RIPv2 on the interface.	**ip summary-address rip** *prefix-address*	-

Configure the RIP Metric Offset

By default, RIP applies the route metric advertised by the neighbor device to the received routes. To modify the metric in some special application scenarios, you can configure the RIP metric offset to correct the metric of the specified route.

If the metric in the incoming direction is configured, RIP modifies the metric of the received routes and saves the routes into the route table. When RIP advertises a metric to the neighbor devices, it advertises the new metric. If the metric in the outgoing direction is configured, the metric is modified only when RIP advertises a metric to the neighbor devices.

Table 4-10 Configure the RIP metric offset

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the RIP configuration mode.	**router rip**	Mandatory. By default, the RIP process is disabled.
Configure RIP to modify the metric of the specified route.	**offset-list** *access-list-name* { **in** | **out** } *metric-offset* [*interface-name*]	Mandatory. By default, no metric is configured for any interface.

Note:

- Route metric offset supports only matching a standard access list.

Configure RIP Route Filtration

A router can filter the received or advertised routes by configuring an Access Control List (ACL) or prefix list. In receiving RIP routes, you can filter some learnt routes; or in announcing RIP routes, you can filter some routes that are advertised to neighbor devices.

Table 4-11 Configure RIP route filtration

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the RIP configuration mode.	**router rip**	Mandatory. By default, the RIP process is disabled.
Configure the RIP route filtering function.	**distribute-list** { *access-list-name* | **prefix** *prefix-list-name* } { **in | out** } [*interface-name*]	Mandatory. By default, the route filtering function is not configured. During the configuration process of the route filtering function, if no interface is specified, route filtering is enabled for all routes that are received and transmitted by all the interfaces covered by RIP.

Note:

- In filtration based on ACL, only a standard ACL is supported.

Configure Metric of RIP Interface

If an interface is overwritten by an RIP process, the corresponding direct route is generated in the database, with the default metric 1. When the route is in the RIP database or it is advertised to neighbor devices, if the interface is configured with a metric, the interface metric is used as the metric of the route.

If the interface metric is changed, the RIP database immediately updates the corresponding direct route of RIP and advertises the new metric to the neighbor devices.

Table 4-12 Configure the Metric of the RIP interface

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the interface configuration mode.	**interface** *interface-name*	-
Configure the metric of the RIP interface.	**ip rip metric** *metric-value*	Mandatory. By default, the RIP interface metric is 1.

Note:

- Configuring the RIP interface metric affects only the metric of the direct subnet of the interface while it does not affect the metric learned by routes.

Configure the Routing Flag for an RIP Interface

The network administrator can attach tags to some routes. Then, in applying a routing policy, the network administrator can perform route filtering or route property advertisement based on the tags.

Only the routing tags of RIPv2 are supported.

Table 4-13 Configure the routing flag for an RIP interface

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the interface configuration mode.	**interface** *interface-name*	-

Step	Command	Description
Configure a tag for the route of the direct subnet of the interface.	**ip rip tag** *tag-value*	-

Configure the Maximum Number of RIP Load Balancing Entries

This command helps you to control the number of RIP load balancing entries for routing.

Table 4-14 Configure the maximum number of RIP load balancing entries

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the RIP configuration mode.	**router rip**	Mandatory. By default, the RIP process is disabled.
Configure the maximum number of RIP load balancing entries.	**maximum-paths** *max-number*	Optional. By default, the maximum number of RIP load balancing entries is 4.

4.2.4. Configure RIP Network Authentication

RIPv2 supports protocol packet authentication, therefore, it can satisfy the high security requirement of some networks. Currently, plain text authentication and Message Digest 5 (MD5) authentication are supported. Plain text authentication features low security because it transmits plain text. MD5 converts an authentication code into the MD5 code for transmission, ensuring higher security.

Owing to the limit of RIPv2 packets, a packet that advertises a route contains only 16 bytes. Therefore, the length of a plain text authentication string must not exceed 16 bytes. Meanwhile, the MD5 code that is converted from any character string is a standard 16-byte code, meeting the requirement on the string length.

Table 4-15 Configure RIP network authentication

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-

Step	Command	Description
Enter the interface configuration mode.	**interface** *interface-name*	-
Configure RIPv2 network authentication.	**ip rip authentication** { { **key** { 0 | 7 } *key-string* } | { **key-chain** *key-chain-name* } | { **mode** { **text** | **md5** | **sm3**} } }	Mandatory. By default, the IPv2 authentication function is not configured.

Note:

- Before implementing MD5 authentication, pay attention to the following points:
- RIPv1 does not support network authentication.
- RIPv2 supports one authentication mode at a time.
- Key ID must be carried in the MD5 authentication information. If you use the **ip rip authentication key** command to configure a password, the key ID is 1. If you use the **ip rip authentication key-chain** command to configure a password, the key ID is the key ID in Key-chain.
- In obtaining a packet transmit authentication password from Key-chain, select a Key ID in the sequence of from small to large. Therefore, the Key ID with the smallest valid transmit password will be selected.
- In obtaining a packet receive authentication password from Key-chain, select the first valid receive password whose Key ID is equal to or larger than the packet receive Key ID. Therefore, if Key IDs are different for the two ends of authentication, the end with the larger Key ID can pass the authentication while the end with the smaller Key ID fails in the authentication.

4.2.5. Configure RIP Network Optimization

Configuration Condition

Before configuring RIP network optimization, ensure that:

- Interface IP addresses have been configured so that neighbor nodes are reachable at the network layer.
- RIP is enabled.

Configure RIP Timers

RIP does not maintain neighbor relations and it does not support route withdrawn; therefore, the protocol provides four configurable timers to control the network convergence speed. The four timers are: route update timer, router timeout timer, route dampening update timer, and route clear timer.

The route timeout time must be at least three times of the route update time. If no route update packet is received within the route timeout time, the route becomes invalid and it enters a dampening cycle. The length of the dampening cycle is determined by the dampening update time. During the cycle, the route will not be cleared. After the dampening cycle is completed, the route enters the clear cycle. During the cycle, the route can be updated. However, if no route update packet is received during the cycle, the route will be deleted.

Table 4-16 Configuring RIP timers

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the RIP configuration mode.	**router rip**	Mandatory. By default, the RIP process is disabled.
Configure RIP timers.	**timers basic** *update-interval invalid-interval holddown-interval flush-interval*	By default, the RIP update interval is 30s, the valid time for advertisement is 180s, the dampening time is 180s, and the clear time is 240s.

Caution:

- In the same RIP routing domain, the **timer basic** configurations on all the devices must be the same to prevent network flapping.

Configure RIP Split Horizon and Toxicity Reverse of RIP

Split horizon and toxicity reverse are mechanisms that are used to prevent route loops.

1. Configure split horizon.

RIP does not advertise routes that it has learnt from an interface to the interface, preventing routing loops.

Table 4-17 Configure RIP split horizon

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the interface configuration mode.	**interface** *interface-name*	-
Configuring RIP split horizon.	**ip split-horizon**	Mandatory. By default, the split horizon function is disabled.

1. Configure toxicity reverse.

RIP announces routes that have been learnt from an interface to the interface, but the route metric is the maximum number of hops, 16, preventing routing loops.

Table 4-18 Configure RIP toxicity reverse

Step	**Command**	**Description**
Enter the global configuration mode.	**configure terminal**	-
Enter the interface configuration mode.	**interface** *interface-name*	-
Configure RIP toxicity reverse.	**ip split-horizon poisoned**	Optional. By default, the toxicity reverse function is enabled.

Note:

- The split horizon and toxicity reverse functions are valid only for the learnt routes, direct routes in the network covered by RIP, and the redistributed direct and static routes.
- The split horizon function and the toxicity reversion function cannot be used at the same time.

Configure RIP Source Address Check

Through source address check, RIP checks the source addresses of the received packets. RIP processes only the packets whose source addresses meet the requirements. The check items include: the packet source address is in the same network segment as the input interface address; the packet source address matches the peer end address of the Point-to-Point (P2P) interface.

By default, RIP is enabled to check whether the source addresses received through the Ethernet port are in the same network segment as the address of the interface, and this function cannot be cancelled.

Table 4-19 Configure RIP source address check

Step	**Command**	**Description**
Enter the global configuration mode.	**configure terminal**	-
Enter the RIP configuration mode.	**router rip**	Mandatory. By default, the RIP process is disabled.

Step	Command	Description
Configure RIP to start source address check on the P2P interface.	**validate-update-source check-p2p-destination**	Mandatory. By default, the peer address of the P2P interface is not checked.

Configure a Static RIP Neighbor

RIP does not maintain neighbor relations, so it does not have the concept of neighbor. Here the neighbor refers to the neighbor RIP routing device. After a static RIP neighbor is specified, RIP sends RIP packets to the neighbor in unicast mode. The configuration is applied to a network that does not support broadcast or multicast, such as point-to-point links. If the configuration is applied to a broadcast or multicast network, it may cause repeated RIP packets in the network.

Table 4-20 Configure a static RIP neighbor

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the RIP configuration mode.	**router rip**	Mandatory. By default, the RIP process is disabled.
Configure advertisement of routes to a neighbor in unicast mode.	**neighbor** *ip-address*	Mandatory. The parameter *ip-address* is the IP address of the peer direct-connect interface.

Note:

- RIP advertises routes only to the interfaces that it covers, and the passive interface setting cannot prevent an interface from sending packets to its static neighbor.

Configure a Passive RIP Interface

To decrease the network bandwidth consumed by the routing protocol, the dynamic routing protocol uses the passive interface function. RIP receives only route update packets on a passive interface, and it does not send route update packets on the passive interface. In a low-speed network with small bandwidth, the passive interface function and the neighbor function cooperate to effectively reduce interactions of RIP routes.

Table 4-21 Configure a passive RIP interface

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the RIP configuration mode.	**router rip**	Mandatory. By default, the RIP process is disabled.
Configure a passive RIP interface.	**passive-interface** { **default** | *interface-name* }	Mandatory. By default, no passive interface is configured.

Note:

- The passive interface function does not restrain an interface from sending unicast route updates to its neighbor devices. When the passive interface function is used with the **neighbor** command, the function does not restrain an interface from sending unicast route updates to its neighbor devices. This application mode controls a router so that it sends route updates only to some neighbor devices in unicast mode instead of sending route updates to all neighbor devices in broadcast mode (or multicast mode in the case of RIPv2).

Configure RIP to Trigger Updates

After a device receives an RIP update packet, to reduce the possibility of introducing loops owning to route table differences, the device advertises the update packet of the route to its neighbor devices immediately instead of waiting for the update timer to time out before an update. The update trigger mechanism speeds up network convergence.

Table 4-22 Configure RIP to trigger updates

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the interface configuration mode.	**interface** *interface-name*	-
Configure RIP to trigger updates on the interface.	**ip rip triggered**	Optional. By default, the update trigger function is disabled.

Configure an RIP Standby Interface

To speed up backup route convergence, RIP newly supports a backup interface (standby interface) function. On the main route interface of RIP, specify a backup interface for the main interface. In a specific application environment, RIP learns RIP routes only from one line, and the backup line does not provide routing information interaction. If the main interface gets offline, RIP sends Request packets to the peer end through the backup interface periodically (Default: 1s) to request for all routes. If the backup interface receives a Response packet from the peer route, RIP cancels sending of Request packets. It updates the local route table, and advertises the local route table to the backup interface. If the backup interface fails to receive a Response packet from the peer end before timeout, RIP cancels sending of Request packets.

Table 4-23 Configure an RIP backup interface

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the interface configuration mode.	**interface** *interface-name*	-
Configure an RIP backup interface.	**ip rip standby** *interface-name* [**timeout** *timeout-value*]	Optional. By default, the backup interface function is disabled, and the default *timeout-value* is 300s.

4.2.6. Configure RIP to Coordinate with BFD

A backup interface can be used only in a specific application environment, but it cannot meet the real-time backup requirement. At this time, RIP provides the point-to-point Bidirectional Forwarding Detection (BFD) function to realize fast convergence and switchover of routes. BFD provides a method for quickly detecting the status of a line between two devices. When BFD detection is enabled between two adjacent RIP devices, if the line between the two devices is faulty, BFD can quickly find the fault and notify RIP. RIP then deletes the RIP route that is associated with the BFD interface. If the route has a backup route, a switchover to the backup route will be performed in a very short period of time (which is determined by BFD settings). Currently, RIP only supports single-hop bi-directional BFD detection.

Table 4-24 Configure RIP to coordinate with BFD

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-

Step	Command	Description
Enter the RIP configuration mode.	**router rip**	Mandatory. By default, the RIP process is disabled.
Enable the BFD function on all the interfaces that are covered by the RIP process.	**bfd all-interfaces**	Mandatory. By default, the BFD function is disabled on all the interfaces that are covered by the RIP process.
Return to the global configuration mode.	**exit**	-
Enter the interface configuration mode.	**interface** *interface-name*	-
Enable the BFD function on the interface.	**ip rip bfd**	Mandatory. By default, the BFD function is disabled on the interface.

Note:

- For the related configuration of BFD, refer to Reliability- BFD command manual.

4.2.7. RIP Monitoring and Maintaining

Table 4-25 RIP monitoring and maintaining

Command	Description
show ip rip [**vrf** *vrf-name*]	Display the basic information about the RIP protocol.
show ip rip [**vrf** *vrf-name*] **database** [**detail** | *prefix/mask* [[**detail** | **longer-prefixes** [**detail**]]]]	Display the information about the RIP routing database.
show ip rip [**vrf** *vrf-name*] **statistics**	Display the RIP protocol statistics.

Command	Description
show ip rip interface [*interface-name*]	Display the RIP interface information.
clear ip rip [**vrf** *vrf-name*] { **process** | **statistics** }	Clear RIP process and statistics.

4.3. RIP Typical Configuration Example

4.3.1. Configure RIP Version

Network Requirements

- RIPv2 runs between Device1 and Device2 for route interaction.

Network Topology

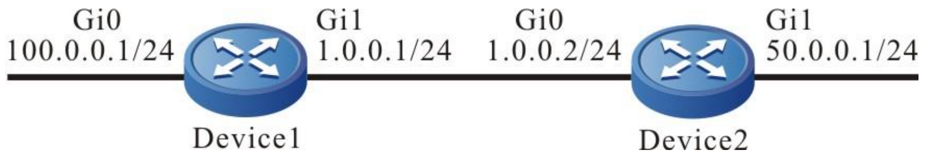

Figure 4-1 Networking for configuring the RIP version

Configuration Steps

Step 1: Configure the IP addresses of the interfaces. (Omitted)

Step 2: Configure RIP.

#Configure Device1.

```
Device1#configure terminal
Device1(config)#router rip
Device1(config-rip)#network 1.0.0.0
Device1(config-rip)#network 100.0.0.0
Device1(config-rip)#exit
```

#Configure Device2.

```
Device2#configure terminal
Device2(config)#router rip
Device2(config-rip)#network 1.0.0.0
Device2(config-rip)#network 50.0.0.0
Device2(config-rip)#exit
```

#Query the route table of Device1.

```
Device1#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external
```

```
C   1.0.0.0/24 is directly connected, 00:23:06, gigabitethernet1
L   1.0.0.1/32 is directly connected, 00:23:06, gigabitethernet1
R   50.0.0.0/8 [120/1] via 1.0.0.2, 00:13:26, gigabitethernet1
C   100.0.0.0/24 is directly connected, 00:23:06, gigabitethernet0
L   100.0.0.1/32 is directly connected, 00:23:06, gigabitethernet0
C   127.0.0.0/8 is directly connected, 76:51:00, lo0
L   127.0.0.1/32 is directly connected, 76:51:00, lo0
```

#Query the route table of Device2.

```
Device2#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

C   1.0.0.0/24 is directly connected, 00:23:06, gigabitethernet0
L   1.0.0.2/32 is directly connected, 00:23:06, gigabitethernet0
C   50.0.0.0/24 is directly connected, 00:23:06, gigabitethernet1
L   50.0.0.1/32 is directly connected, 00:23:06, gigabitethernet1
R   100.0.0.0/8 [120/1] via 1.0.0.1, 00:13:26, gigabitethernet0
C   127.0.0.0/8 is directly connected, 76:51:00, lo0
L   127.0.0.1/32 is directly connected, 76:51:00, lo0
```

According to the route table, the route advertised by the device uses a 8-bit natural mask.

Step 3: Configure the RIP version.

#Configure Device1.

```
Device1(config)#router rip
Device1(config-rip)#version 2
Device1(config-rip)#exit
```

#Configure Device2.

```
Device2(config)#router rip
Device2(config-rip)#version 2
Device2(config-rip)#exit
```

Step 4: Check the result.

#Query the route table of Device1.

```
Device1#show ip route
```

```
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

C  1.0.0.0/24 is directly connected, 00:23:06, gigabitethernet1
L  1.0.0.1/32 is directly connected, 00:23:06, gigabitethernet1
R  50.0.0.0/24 [120/1] via 1.0.0.2, 00:13:26, gigabitethernet1
C  100.0.0.0/24 is directly connected, 00:23:06, gigabitethernet0
L  100.0.0.1/32 is directly connected, 00:23:06, gigabitethernet0
C  127.0.0.0/8 is directly connected, 76:51:00, lo0
C  127.0.0.1/32 is directly connected, 76:51:00, lo0
```

#Query the route table of Device2.

```
Device2#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

C  1.0.0.0/24 is directly connected, 00:23:06, gigabitethernet0
L  1.0.0.2/32 is directly connected, 00:23:06, gigabitethernet0
C  50.0.0.0/24 is directly connected, 00:23:06, gigabitethernet1
L  50.0.0.1/32 is directly connected, 00:23:06, gigabitethernet1
R  100.0.0.0/24 [120/1] via 1.0.0.1, 00:13:26, gigabitethernet0
C  127.0.0.0/8 is directly connected, 76:51:00, lo0
L  127.0.0.1/32 is directly connected, 76:51:00, lo0
```

According to the route table, the route advertised by the device uses a 24-bit accurate mask.

4.3.2. Configure RIP to Redistribute Routes

Network Requirements

- OSPF runs between Device1 and Device2. Device2 learns OSPF routes 100.0.0.0/24 and 200.0.0.0/24 advertised by Device1.
- RIPv2 runs between Device2 and Device3. Device2 redistributes OSPF route 100.0.0.0/24 to RIP and advertises the route to Device3.

Network Topology

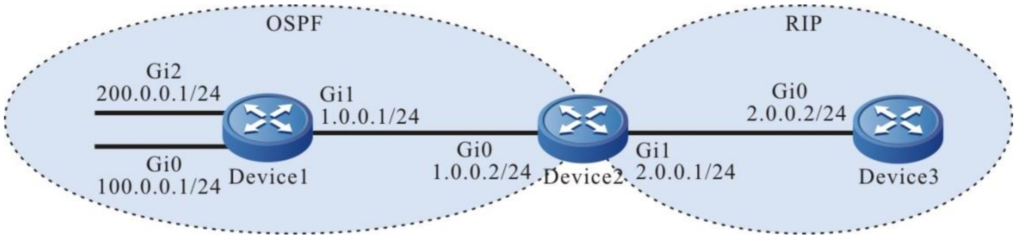

Figure 4-2 Networking for configuring RIP to redistribute routes

Configuration Steps

Step 1: Configure the IP addresses of the interfaces. (Omitted)

Step 2: Configure OSPF.

#Configure Device1.

```
Device1#configure terminal
Device1(config)#router ospf 100
Device1(config-ospf)#network 1.0.0.0 0.0.0.255 area 0
Device1(config-ospf)#network 100.0.0.0 0.0.0.255 area 0
Device1(config-ospf)#network 200.0.0.0 0.0.0.255 area 0
Device1(config-ospf)#exit
```

#Configure Device2.

```
Device2#configure terminal
Device2(config)#router ospf 100
Device2(config-ospf)#network 1.0.0.0 0.0.0.255 area 0
Device2(config-ospf)#exit
```

#Query the route table of Device2.

```
Device2#show ip route ospf
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
     U - Per-user Static route
     O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external


O   100.0.0.0/24 [110/2] via 1.0.0.1, 00:04:12, gigabitethernet0
O   200.0.0.0/24 [110/2] via 1.0.0.1, 00:04:12, gigabitethernet0
```

According to the route table, Device2 has learnt the OSPF routes that have been advertised by Device1.

Step 3: Configure RIP.

#Configure Device2.

```
Device2(config)#router rip
Device2(config-rip)#version 2
```

```
Device2(config-rip)#network 2.0.0.0
Device2(config-rip)#exit
```

#Configure Device3.

```
Device3#configure terminal
Device3(config)#router rip
Device3(config-rip)#version 2
Device3(config-rip)#network 2.0.0.0
Device3(config-rip)#exit
```

Step 4: Configure the routing policy.

#On Device2, configure route-map to invoke ACL to match 100.0.0.0/24 and filter 200.0.0.0/24.

```
Device2(config)#ip access-list standard 1
Device2(config-std-nacl)#permit 100.0.0.0 0.0.0.255
Device2(config-std-nacl)#exit
Device2(config)#route-map OSPFtoRIP
Device2(config-route-map)#match ip address 1
Device2(config-route-map)#exit
```

Note:

- In configuring a routing policy, you can create a filtration rule based on a prefix list or ACL. The prefix list can precisely match routing masks while the ACL cannot match routing masks.

Step 5: Configure RIP to redistribute routes.

#Configure Device2.

```
Device2(config)#router rip
Device2(config-rip)#redistribute ospf 100 route-map OSPFtoRIP
Device2(config-rip)#exit
```

Step 6: Check the result.

#Query the RIP route table of Device2.

```
Device2#show ip rip database
Types: N - Network, L - Learn, R - Redistribute, D - Default config, S - Static config
Proto: C - connected, S - static, R - RIP, O - OSPF, E - IRMP,
    o - SNSP, B - BGP, i-ISIS

RIP routing database in VRF kernel (Counter 3):
T/P Network        ProID Metric Next-Hop      From         Time  Tag   Interface
N/C 2.0.0.0/24      none  1      --            --           --    0    gigabitethernet1
R/O 100.0.0.0/24     1    1      1.0.0.1       --           --    0    gigabitethernet0
```

#Query the route table of Device3.

```
Device3#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
    U - Per-user Static route
    O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

C   2.0.0.0/24 is directly connected, 00:23:06, gigabitethernet0
L   2.0.0.2/32 is directly connected, 00:23:06, gigabitethernet0
R   100.0.0.0/24 [120/1] via 2.0.0.1, 00:13:26, gigabitethernet0
C   127.0.0.0/8 is directly connected, 76:51:00, lo0
L   127.0.0.1/32 is directly connected, 76:51:00, lo0
```

By querying the RIP route table on Device2 and the querying the route table on Device3, it is found that route 100.0.0.0/24 on Device2 has been redistributed to RIP and route 200.0.0.0/24 has been successfully filtered out.

Caution:

- In an actual application, if there are two or more AS boundary routers, it is recommended that you do not redistribute routes between different routing protocols. If route redistribution must be configured, you are required to configure route control policies such as route filtering and filtration summary on the AS boundary routers to prevent routing loops.

4.3.3. Configure RIP Metric Offset

Network Requirements

- RIPv2 runs between Device1, Device2, Device3, and Device4.
- Device1 learns route 200.0.0.0/24 from both Device2 and Device3.
- On Device1, set the route metric offset in the receive direction so that Device1 selects the route advertised by Device2 with priority.

Network Topology

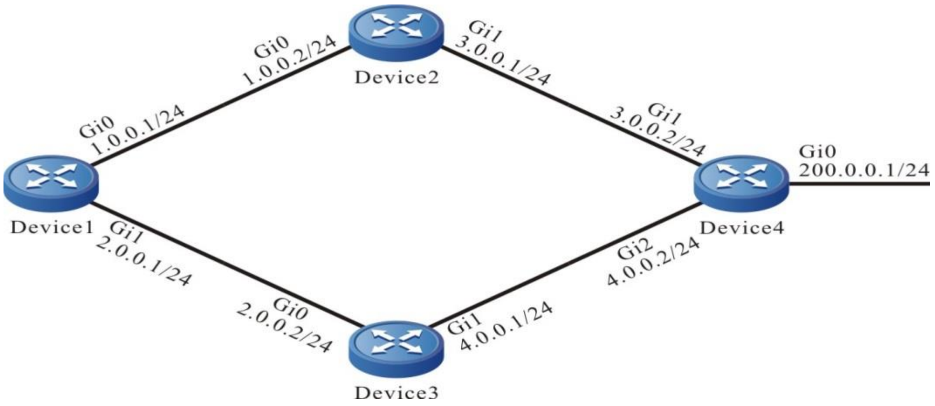

Figure 4-3 Networking for configuring the RIP metric offset

Configuration Steps

Step 1: Configure the IP addresses of the interfaces. (Omitted)

Step 2: Configure RIP.

#Configure Device1.

```
Device1#configure terminal
Device1(config)#router rip
Device1(config-rip)#version 2
Device1(config-rip)#network 1.0.0.0
Device1(config-rip)#network 2.0.0.0
Device1(config-rip)#exit
```

#Configure Device2.

```
Device2#configure terminal
Device2(config)#router rip
Device2(config-rip)#version 2
Device2(config-rip)#network 1.0.0.0
Device2(config-rip)#network 3.0.0.0
Device2(config-rip)#exit
```

#Configure Device3.

```
Device3#configure terminal
Device3(config)#router rip
Device3(config-rip)#version 2
Device3(config-rip)#network 2.0.0.0
Device3(config-rip)#network 4.0.0.0
Device3(config-rip)#exit
```

#Configure Device4.

```
Device4#configure terminal
Device4(config)#router rip
Device4(config-rip)#version 2
Device4(config-rip)#network 3.0.0.0
Device4(config-rip)#network 4.0.0.0
Device4(config-rip)#network 200.0.0.0
Device4(config-rip)#exit
```

#Query the route table of Device1.

```
Device1#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
     U - Per-user Static route
     O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external
```

```
C   1.0.0.0/24 is directly connected, 00:23:06, gigabitethernet0
L   1.0.0.1/32 is directly connected, 00:23:06, gigabitethernet0
C   2.0.0.0/24 is directly connected, 00:22:56, gigabitethernet1
L   2.0.0.1/32 is directly connected, 00:22:56, gigabitethernet1
R   3.0.0.0/24 [120/1] via 1.0.0.2, 00:13:26, gigabitethernet0
R   4.0.0.0/24 [120/1] via 2.0.0.2, 00:11:04, gigabitethernet1
C   127.0.0.0/8 is directly connected, 76:51:00, lo0
L   127.0.0.1/32 is directly connected, 76:51:00, lo0
R   200.0.0.0/24 [120/2] via 1.0.0.2, 00:08:31, gigabitethernet0
                 [120/2] via 2.0.0.2, 00:08:31, gigabitethernet1
```

According to the route table of Device1, two routes to 200.0.0.0/24 are available.

Step 3: Configure the ACL.

#Configure Device1.

```
Device1(config)#ip access-list standard 1
Device1(config-std-nacl)#permit 200.0.0.0 0.0.0.255
Device1(config-std-nacl)#exit
```

Step 4: Configure a metric offset.

#On Device1, configure the metric offset list and increase the metric of the route that has been learnt from interface gigabitethernet1 and matches ACL to 3.

```
Device1(config)#router rip
Device1(config-rip)#offset-list 1 in 3 gigabitethernet1
Device1(config-rip)#exit
```

Step 5: Check the result.

#Query the route table of Device1.

```
Device1#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

C   1.0.0.0/24 is directly connected, 00:33:59, gigabitethernet0
L   1.0.0.1/32 is directly connected, 00:33:59, gigabitethernet0
C   2.0.0.0/24 is directly connected, 00:33:50, gigabitethernet1
L   2.0.0.1/32 is directly connected, 00:33:50, gigabitethernet1
R   3.0.0.0/24 [120/1] via 1.0.0.2, 00:24:20, gigabitethernet0
```

```
R   4.0.0.0/24 [120/1] via 2.0.0.2, 00:21:57, gigabitethernet1
C   127.0.0.0/8 is directly connected, 77:01:54, lo0
L   127.0.0.1/32 is directly connected, 77:01:54, lo0
R   200.0.0.0/24 [120/2] via 1.0.0.2, 00:19:25, gigabitethernet0
```

According to the route table of Device1, the next-hop output interface of route 200.0.0.0/24 is only gigabitethernet0, indicating that Device1 has selected the route advertised by Device2 with priority.

Note:

- The route metric offset list can be applied to all interfaces or a specified interface, and it can be used in both the receive and advertisement directions.

4.3.4. Configure RIP Route Filtration

Network Requirements

- RIPv2 runs between Device1 and Device2 for route interaction.
- Device1 learns two routes 2.0.0.0/24 and 3.0.0.0/24 that have been advertised by Device2, and then it filters route 3.0.0.0/24 in the advertisement direction of Device2.

Network Topology

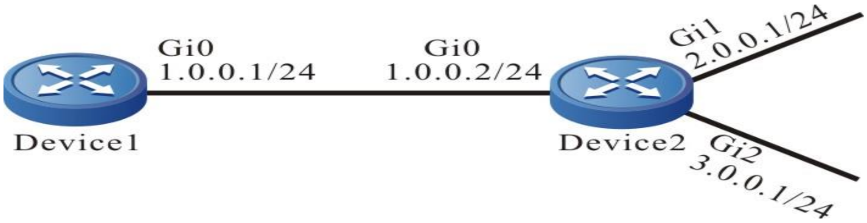

Figure 4-4 Networking for configuring RIP Route filtration

Configuration Steps

Step 1: Configure the IP addresses of the interfaces. (Omitted)

Step 2: Configure RIP.

#Configure Device1.

```
Device1#configure terminal
Device1(config)#router rip
Device1(config-rip)#version 2
Device1(config-rip)#network 1.0.0.0
Device1(config-rip)#exit
```

#Configure Device2.

```
Device2#configure terminal
Device2(config)#router rip
Device2(config-rip)#version 2
Device2(config-rip)#network 1.0.0.0
Device2(config-rip)#network 2.0.0.0
Device2(config-rip)#network 3.0.0.0
```

```
Device2(config-rip)#exit
```

#Query the route table of Device1.

```
Device1#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

C   1.0.0.0/24 is directly connected, 00:23:06, gigabitethernet0
L   1.0.0.1/32 is directly connected, 00:23:06, gigabitethernet0
R   2.0.0.0/24 [120/1] via 1.0.0.2, 00:13:26, gigabitethernet0
R   3.0.0.0/24 [120/1] via 1.0.0.2, 00:13:26, gigabitethernet0
C   127.0.0.0/8 is directly connected, 76:51:00, lo0
L   127.0.0.1/32 is directly connected, 76:51:00, lo0
```

According to the route table, Device1 has learnt two routes advertised by Device2.

Step 3: Configure the ACL.

#Configure Device2.

```
Device2(config)#ip access-list standard 1
Device2(config-std-nacl)#permit 2.0.0.0 0.0.0.255
Device2(config-std-nacl)#exit
```

Note:

- In configuring route filtration, you can create a filtration rule based on a prefix list or ACL. The prefix list can precisely match routing masks while the ACL cannot match routing masks.

Step 4: Configure route filtration.

#Configure route filtering in the output direction of interface gigabitethernet0 of Device2.

```
Device2(config)#router rip
Device2(config-rip)#distribute-list 1 out gigabitethernet0
Device2(config-rip)#exit
```

Step 5: Check the result.

#Query the route table of Device1.

```
Device1#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external
```

```
C   1.0.0.0/24 is directly connected, 00:23:06, gigabitethernet0
L   1.0.0.1/32 is directly connected, 00:23:06, gigabitethernet0
R   2.0.0.0/24 [120/1] via 1.0.0.2, 00:13:26, gigabitethernet0
C   127.0.0.0/8 is directly connected, 76:51:00, lo0
L   127.0.0.1/32 is directly connected, 76:51:00, lo0
```

According to the route table, Device2 does not advertise route 3.0.0.0/24 to Device1, but the route is deleted from the route table of Device only after the route times out.

Note:

- The **distribute-list** can be applied to all interfaces or a specified interface, and it can be used in both the receive and advertisement directions.

4.3.5. Configure RIP Route Summary

Network Requirements

- RIPv2 runs between Device1, Device2, Device3, and Device4 for route interaction.
- Device1 learns two routes 100.1.0.0/24 and 100.2.0.0/24 from Device2. To reduce the size of the route table of Device1, it is required that Device advertises only the route summary of the two route to Device1.

Network Topology

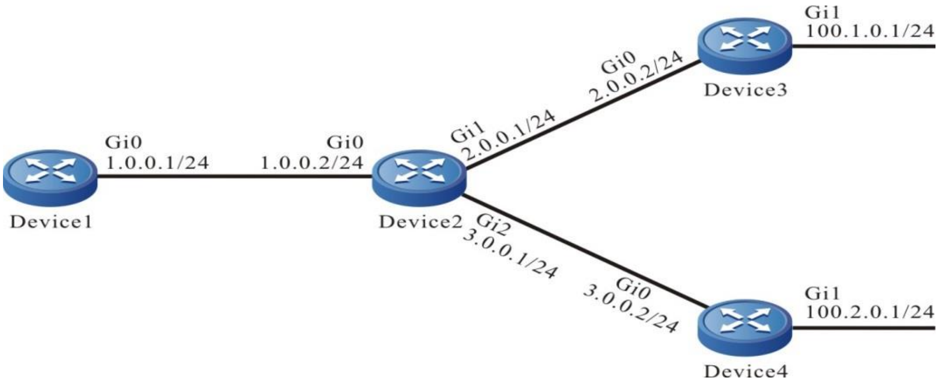

Figure 4-5 Networking for configuring RIP route summary

Configuration Steps

Step 1: Configure the IP addresses of the interfaces. (Omitted)

Step 2: Configure RIP.

#Configure Device1.

```
Device1#configure terminal
Device1(config)#router rip
Device1(config-rip)#version 2
Device1(config-rip)#network 1.0.0.0
Device1(config-rip)#exit
```

#Configure Device2.

```
Device2#configure terminal
Device2(config)#router rip
```

```
Device2(config-rip)#version 2
Device2(config-rip)#network 1.0.0.0
Device2(config-rip)#network 2.0.0.0
Device2(config-rip)#network 3.0.0.0
Device2(config-rip)#exit
```

#Configure Device3.

```
Device3#configure terminal
Device3(config)#router rip
Device3(config-rip)#version 2
Device3(config-rip)#network 2.0.0.0
Device3(config-rip)#network 100.0.0.0
Device3(config-rip)#exit
```

#Configure Device4.

```
Device4#configure terminal
Device4(config)#router rip
Device4(config-rip)#version 2
Device4(config-rip)#network 3.0.0.0
Device4(config-rip)#network 100.0.0.0
Device4(config-rip)#exit
```

#Query the route table of Device1.

```
Device1#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

C   1.0.0.0/24 is directly connected, 00:23:06, gigabitethernet0
L   1.0.0.1/32 is directly connected, 00:23:06, gigabitethernet0
R   2.0.0.0/24 [120/1] via 1.0.0.2, 00:13:26, gigabitethernet0
R   3.0.0.0/24 [120/1] via 1.0.0.2, 00:13:26, gigabitethernet0
R   100.1.0.0/24 [120/2] via 1.0.0.2, 00:08:31, gigabitethernet0
R   100.2.0.0/24 [120/2] via 1.0.0.2, 00:08:31, gigabitethernet0
C   127.0.0.0/8 is directly connected, 76:51:00, lo0
L   127.0.0.1/32 is directly connected, 76:51:00, lo0
```

Step 3: Configure a summary of routes on a interface.

#On Device2, configure a route summary 100.0.0.0/8.

```
Device2(config)#interface gigabitethernet0
Device2(config-if-gigabitethernet0)#ip summary-address rip 100.0.0.0/8
Device2(config-if-gigabitethernet0)#exit
```

Step 4: Check the result.

#Query the route table of Device1.

```
Device1#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

C   1.0.0.0/24 is directly connected, 00:24:06, gigabitethernet0
L   1.0.0.1/32 is directly connected, 00:24:06, gigabitethernet0
R   2.0.0.0/24 [120/1] via 1.0.0.2, 00:14:26, gigabitethernet0
R   3.0.0.0/24 [120/1] via 1.0.0.2, 00:14:26, gigabitethernet0
R   100.0.0.0/8 [120/2] via 1.0.0.2, 00:00:31, gigabitethernet0
C   127.0.0.0/8 is directly connected, 76:51:00, lo0
L   127.0.0.1/32 is directly connected, 76:51:00, lo0
```

On Device1, the route summary 100.0.0.0/8 advertised by Device2 and learnt by Device1 is displayed. The two routes that are contained in the route summary can be deleted only after timeout.

Note:

- RIP supports global auto route summary and interface manual route summary. In RIPv2, the global auto route summary function is disabled.

4.3.6. Configure RIP to Coordinate with BFD

Network Requirements

- RIPv2 runs between Device1, Device2, and Device3 for route interaction.
- Device1 learns route 3.0.0.0/24 from Device2 and Device3. Then, configure route metric offset so that Device1 selects the route advertised by Device2 with priority. At this time, the line between Device1 and Device2 becomes the main line of the route. The line between Device1 and Device3 becomes and backup line of the route.
- Configure BFD between Device1 and Device2. When the line between Device1 and Device2 becomes faulty, configure RIP to coordinate with BFD between Device1 and Device2 to quickly detect line faults. When BFD finds a main line failure, it triggers an RIP route update. Then the route 3.0.0.0/24 is switched over to the backup line.

Network Topology

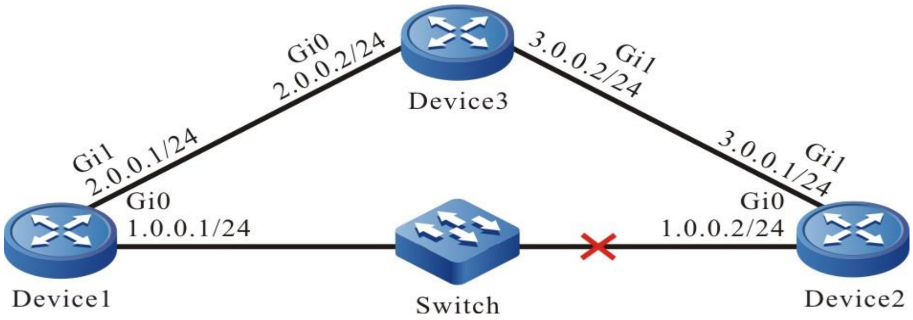

Figure 4-6 Networking for configuring RIP to coordinate with BFD

Configuration Steps

Step 1: Configure the IP addresses of the interfaces. (Omitted)

Step 2: Configure RIP.

#Configure Device1.

```
Device1#configure terminal
Device1(config)#router rip
Device1(config-rip)#version 2
Device1(config-rip)#network 1.0.0.0
Device1(config-rip)#network 2.0.0.0
Device1(config-rip)#exit
```

#Configure Device2.

```
Device2#configure terminal
Device2(config)#router rip
Device2(config-rip)#version 2
Device2(config-rip)#network 1.0.0.0
Device2(config-rip)#network 3.0.0.0
Device2(config-rip)#exit
```

#Configure Device3.

```
Device3#configure terminal
Device3(config)#router rip
Device3(config-rip)#version 2
Device3(config-rip)#network 2.0.0.0
Device3(config-rip)#network 3.0.0.0
Device3(config-rip)#exit
```

#Query the route table of Device1.

```
Device1#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

C  1.0.0.0/24 is directly connected, 01:30:23, gigabitethernet0
L  1.0.0.1/32 is directly connected, 01:30:23, gigabitethernet0
C  2.0.0.0/24 is directly connected, 01:30:14, gigabitethernet1
L  2.0.0.1/32 is directly connected, 01:30:14, gigabitethernet1
R  3.0.0.0/24 [120/1] via 1.0.0.2, 01:20:44, gigabitethernet0
              [120/1] via 2.0.0.2, 00:00:02, gigabitethernet1
C  127.0.0.0/8 is directly connected, 77:58:18, lo0
L  127.0.0.1/32 is directly connected, 77:58:18, lo0
```

Device1 has learnt route 3.0.0.0/24 from both Device2 and Device3.

Step 3: Configure a route metric offset.

#On Device1, configure a route metric offset at the input direction of interface gigabitethernet1 so that the metric of the routes that match ACL is increased by 3.

```
Device1(config)#ip access-list standard 1
Device1(config-std-nacl)#permit 3.0.0.0 0.0.0.255
Device1(config)#exit
Device1(config)#router rip
Device1(config-rip)#offset-list 1 in 3 gigabitethernet1
Device1(config-rip)#exit
```

#Query the route table of Device1.

```
Device1#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

C  1.0.0.0/24 is directly connected, 01:30:23, gigabitethernet0
L  1.0.0.1/32 is directly connected, 01:30:23, gigabitethernet0
C  2.0.0.0/24 is directly connected, 01:30:14, gigabitethernet1
L  2.0.0.1/32 is directly connected, 01:30:14, gigabitethernet1
R  3.0.0.0/24 [120/1] via 1.0.0.2, 01:20:44, gigabitethernet0
C  127.0.0.0/8 is directly connected, 77:58:18, lo0
L  127.0.0.1/32 is directly connected, 77:58:18, lo0
```

After a route metric offset is configured, Device1 selects route 3.0.0.0/24 advertised by Device2.

Step 4: Configure BFD.

#Configure Device1.

```
Device1(config)#interface gigabitethernet0
Device1(config-if-gigabitethernet0)#ip rip bfd
Device1(config-if-gigabitethernet0)#exit
```

#Configure Device2.

```
Device2(config)#interface gigabitethernet0
Device2(config-if-gigabitethernet0)#ip rip bfd
Device2(config-if-gigabitethernet0)#exit
```

Step 5: Check the result.

#On Device1, query the BFD information.

```
Device1#show bfd session
OurAddr            NeighAddr          LD/RD          State       Holddown
interface
1.0.0.1        1.0.0.2         302/504      UP       5000       gigabitethernet0
```

#If the line between Device1 and Device2 becomes faulty, the route can quickly switch over to the backup line.

#On Device1, query the route information.

```
Device1#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

C   2.0.0.0/24 is directly connected, 02:07:47, gigabitethernet1
L   2.0.0.1/32 is directly connected, 02:07:47, gigabitethernet1
R   3.0.0.0/24 [120/4] via 2.0.0.2, 00:01:14, gigabitethernet1
C   127.0.0.0/8 is directly connected, 78:35:51, lo0
L   127.0.0.1/32 is directly connected, 78:35:51, lo0
```

4.3.7. Configure RIP Backup Interface

Network Requirements

- RIPv2 runs between Device1, Device2, and Device3 for route interaction.
- Device1 learns route 3.0.0.0/24 from Device2 and Device3. Then, configure route metric offset so that Device1 selects the route advertised by Device2 with priority. At this time, the line between Device1 and Device2 becomes the main line of the route. The line between Device1 and Device3 becomes and backup line of the route.
- On Device1, configure an RIP backup interface. If the main line is normal, the route passes the main line. If the main line is faulty, the route quickly switches to the backup line.

Network Topology

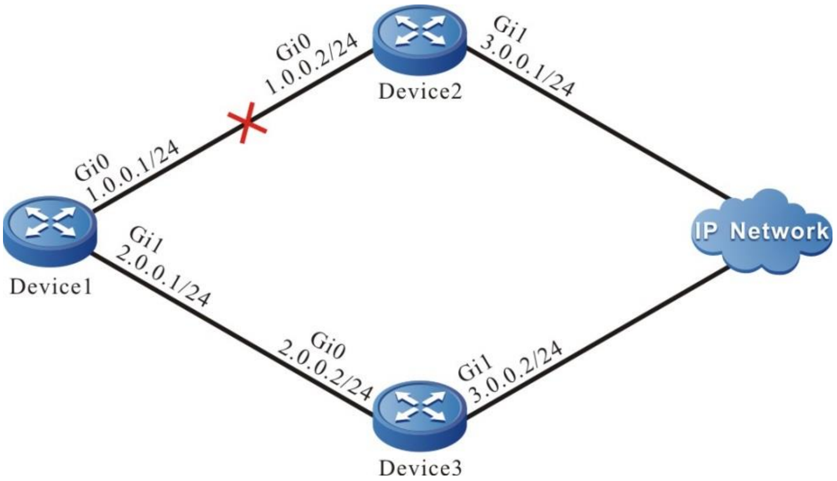

Figure 4-7 Networking for configuring an RIP backup interface

Configuration Steps

Step 1: Configure the IP addresses of the interfaces. (Omitted)

Step 2: Configure RIP.

#Configure Device1.

```
Device1#configure terminal
Device1(config)#router rip
Device1(config-rip)#version 2
Device1(config-rip)#network 1.0.0.0
Device1(config-rip)#network 2.0.0.0
Device1(config-rip)#exit
```

#Configure Device2.

```
Device2#configure terminal
Device2(config)#router rip
Device2(config-rip)#version 2
Device2(config-rip)#network 1.0.0.0
Device2(config-rip)#network 3.0.0.0
Device2(config-rip)#exit
```

#Configure Device3.

```
Device3#configure terminal
Device3(config)#router rip
Device3(config-rip)#version 2
Device3(config-rip)#network 2.0.0.0
Device3(config-rip)#network 3.0.0.0
Device3(config-rip)#exit
```

#Query the route table of Device1.

```
Device1#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

C  1.0.0.0/24 is directly connected, 01:30:23, gigabitethernet0
L  1.0.0.1/21 is directly connected, 01:30:23, gigabitethernet0
C  2.0.0.0/24 is directly connected, 01:30:14, gigabitethernet1
L  2.0.0.1/32 is directly connected, 01:30:14, gigabitethernet1
R  3.0.0.0/24 [120/1] via 1.0.0.2, 01:20:44, gigabitethernet0
              [120/1] via 2.0.0.2, 00:00:02, gigabitethernet1
C  127.0.0.0/8 is directly connected, 77:58:18, lo0
L  127.0.0.1/32 is directly connected, 77:58:18, lo0
```

Device1 has learnt route 3.0.0.0/24 from both Device2 and Device3.

Step 3: Configure a route metric offset.

#On Device1, configure a route metric offset at the input direction of interface gigabitethernet1 so that the metric of the routes that match ACL is increased by 3.

```
Device1(config)#ip access-list standard 1
Device1(config-std-nacl)#permit 3.0.0.0 0.0.0.255
Device1(config)#exit
Device1(config)#router rip
Device1(config-rip)#offset-list 1 in 3 gigabitethernet1
Device1(config-rip)#exit
```

#Query the route table of Device1.

```
Device1#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

C  1.0.0.0/24 is directly connected, 01:30:23, gigabitethernet0
L  1.0.0.1/32 is directly connected, 01:30:23, gigabitethernet0
C  2.0.0.0/24 is directly connected, 01:30:14, gigabitethernet1
L  2.0.0.1/32 is directly connected, 01:30:14, gigabitethernet1
R  3.0.0.0/24 [120/1] via 1.0.0.2, 01:20:44, gigabitethernet0
C  127.0.0.0/8 is directly connected, 77:58:18, lo0
L  127.0.0.1/32 is directly connected, 77:58:18, lo0
```

After the route metric offset is configured, Device1 selects route 3.0.0.0/24 advertised by Device2.

Step 4: Configure a backup interface.

#On Device1, configure interface gigabitethernet1 as the RIP backup interface of gigabitethernet0.

```
Device1(config)#interface gigabitethernet0
Device1(config-if-gigabitethernet0)#ip rip standby gigabitethernet1
Device1(config-if-gigabitethernet0)#exit
```

Step 5: Check the result.

#If the line between Device1 and Device2 becomes faulty, the route can quickly switch over to the backup line between Device1 and Device3.

#On Device1, query the route information.

```
Device1#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
     U - Per-user Static route
     O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

C   2.0.0.0/24 is directly connected, 02:07:47, gigabitethernet1
L   2.0.0.1/32 is directly connected, 02:07:47, gigabitethernet1
R   3.0.0.0/24 [120/4] via 2.0.0.2, 00:01:14, gigabitethernet1
C   127.0.0.0/8 is directly connected, 78:35:51, lo0
L   127.0.0.1/32 is directly connected, 78:35:51, lo0
```

4.3.8. Configure Passive RIP Interface

Network Requirements

- RIPv2 runs between Device1 and Device2 for route interaction.
- On Device1, configure a passive interface which does not send update packets to Device2.

Network Topology

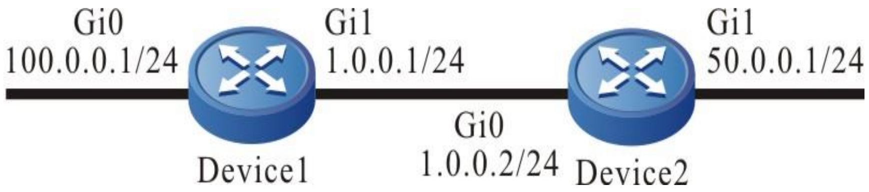

Figure 4-8 Networking for configuring an RIP passive interface

Configuration Steps

Step 1: Configure the IP addresses of the interfaces. (Omitted)

Step 2: Configure RIP.

#Configure Device1.

```
Device1#configure terminal
Device1(config)#router rip
Device1(config-rip)#version 2
Device1(config-rip)#network 1.0.0.0
Device1(config-rip)#network 100.0.0.0
Device1(config-rip)#exit
```

#Configure Device2.

```
Device2#configure terminal
Device2(config)#router rip
Device2(config-rip)#version 2
Device2(config-rip)#network 1.0.0.0
Device2(config-rip)#network 50.0.0.0
Device2(config-rip)#exit
```

#Query the route table of Device1.

```
Device1#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
    U - Per-user Static route
    O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

C  1.0.0.0/24 is directly connected, 00:23:06, gigabitethernet1
L  1.0.0.1/32 is directly connected, 00:23:06, gigabitethernet1
R  50.0.0.0/24 [120/1] via 1.0.0.2, 00:13:26, gigabitethernet1
C  100.0.0.0/24 is directly connected, 00:23:06, gigabitethernet0
L  100.0.0.1/32 is directly connected, 00:23:06, gigabitethernet0
C  127.0.0.0/8 is directly connected, 76:51:00, lo0
L  127.0.0.1/32 is directly connected, 76:51:00, lo0
```

#Query the route table of Device2.

```
Device2#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
    U - Per-user Static route
    O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

C  1.0.0.0/24 is directly connected, 00:23:06, gigabitethernet0
L  1.0.0.2/32 is directly connected, 00:23:06, gigabitethernet0
C  50.0.0.0/24 is directly connected, 00:23:06, gigabitethernet1
L  50.0.0.1/32 is directly connected, 00:23:06, gigabitethernet1
```

```
R   100.0.0.0/24 [120/1] via 1.0.0.1, 00:13:26, gigabitethernet0
C   127.0.0.0/8 is directly connected, 76:51:00, lo0
L   127.0.0.1/32 is directly connected, 76:51:00, lo0
```

Step 3: Configure a passive interface.

#Configure Device1.

```
Device1(config)#router rip
Device1(config-rip)#passive-interface gigabitethernet1
Device1(config-rip)#exit
```

gigabitethernet1 of Device1 is configured as a passive interface which does not send update packets to Device2, but Device2 can still receive update packets.

Step 4: Check the result.

#Query the route table of Device1.

```
Device1#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

C   1.0.0.0/24 is directly connected, 00:23:06, gigabitethernet1
L   1.0.0.1/32 is directly connected, 00:23:06, gigabitethernet1
R   50.0.0.0/24 [120/1] via 1.0.0.2, 00:13:26, gigabitethernet1
C   100.0.0.0/24 is directly connected, 00:23:06, gigabitethernet0
L   100.0.0.1/32 is directly connected, 00:23:06, gigabitethernet0
C   127.0.0.0/8 is directly connected, 76:51:00, lo0
L   127.0.0.1/32 is directly connected, 76:51:00, lo0
```

Route 50.0.0.0/24 is still kept on Device1. On Device2, after the RIP route times out and is deleted, route 100.0.0.0/24 is deleted from the route table.

#Query the route table of Device2.

```
Device2#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

C   1.0.0.0/24 is directly connected, 00:25:06, gigabitethernet0
L   1.0.0.2/32 is directly connected, 00:25:06, gigabitethernet0
C   50.0.0.0/24 is directly connected, 00:25:06, gigabitethernet1
```

```
L   50.0.0.2/32 is directly connected, 00:25:06, gigabitethernet1
C   127.0.0.0/8 is directly connected, 77:51:00, lo0
L   127.0.0.1/32 is directly connected, 77:51:00, lo0
```

5. RIPNG

5.1. Overview

RIPng, also known as next-generation RIP protocol, is a dynamic routing protocol used by the IPv6 networks to provide routing information for the IPv6 packet forwarding. RIPng is extended on RIP-2. The working principle of the RIPng protocol is basically the same as that of the RIP protocol. In order to adapt to the IPv6 network, RIPng has made the following changes to the original RIP protocol:

- UDP port number: The RIPng protocol uses UDP port number 521 to send and receive the protocol packets;
- Multicast address: The RIPng protocol uses FF02::9 as the multicast address of the RIPng router in the local range of the link, and does not support broadcasting.
- Prefix length: The destination address of the RIPng protocol route uses the 128-bit prefix length;
- Next-hop address: The RIPng protocol uses 128-bit IPv6 address;
- Source address: The RIPng protocol uses the link local address FE80:/10 as the source address to send the RIPng protocol packet.

The protocol specifications related to RIPng include RFC2080 and RFC2081.

5.2. RIPng Function Configuration

Table 5-1 RIPng function configuration list

Configuration Tasks	
Configure RIPng basic functions	Enables RIPng globally
Configure RIPng route generation	Configure RIPng to advertise the default route.
	Configure RIP to re-distribute routes
Configure RIPng route control	Configure the administrative distance of RIPng
	Configure a RIPng route summary.
	Configure the RIPng metric offset.
	Configure RIPng route filtering
	Configure the metric of the RIPng interface.

Configuration Tasks	
Configure RIPng route control	Configure the routing flag for a RIPng interface.
	Configure the maximum load balancing for RIPng
Configure RIPng network optimization	Configure RIPng timers.
	Configure RIPng split horizon and toxicity reverse of RIP.
	Configure a RIPng static neighbor.
	Configure a RIPng passive interface.

5.2.1. Configure Basic Functions of RIPng

Configuration Condition

Before configuring the basic functions of RIPng, ensure that:

- The link layer protocol has been configured to ensure normal communication at the link layer.
- The IPv6 capability of the interface is enabled.

Enable RIPng Globally

Before using RIPng, make the following configurations:

- Create a RIPng process.
- Enable the RIPng protocol on the interface

Table 5-2 Enabling RIPng globally

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Create a RIPng process and enter the RIPng configuration mode.	**ipv6 router rip** *process-id*	Mandatory. By default, the RIPng process is disabled.

Step	Command	Description
Return to the global configuration mode	**exit**	-
Enter the interface configuration mode	**interface** *interface-name*	-
Enable the RIPng protocol on the interface	**ipv6 rip enable** *process-id*	Mandatory By default, do not enable the RIPng protocol on the interface.

5.2.2. Configure RIPng Route Generation

Configuration Condition

Before configuring RIPng route generation, ensure that:

- The IPv6 capability of the interface is enabled.
- RIPng is enabled.

Configure RIPng to Advertise the Default Route

Through configuration, a device can send the default route in all RIPng interfaces to set itself as the default gateway of other neighbor devices.

Table 5-3 Configure RIPng to advertise the default route

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the RIPng configuration mode.	**ipv6 router rip** *process-id*	Mandatory By default, the RIPng process is disabled.
Configure RIPng to advertise the default route.	**default-information originate** [**metric** *value*]	Mandatory By default, RIPng does not advertise the default route.

Note:

- If a default route (::/0) is learnt, the default route (::/0) advertised by the local device is replaced. When loops exists in a network, network flapping may be caused. In using this command, prevent other devices in the same routing domain from enabling the command at the same time.

Configure RIPng to Redistribute Routes

By redistributing routes, you can introduce the routes generated by other protocols to RIPng.

Table 5-4 Configure RIPng to redistribute routes

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the RIPng configuration mode.	**ipv6 router rip** *process-id*	Mandatory By default, the RIPng process is disabled.
Configure the default metric for the routes of other protocols introduced to RIPng.	**default-metric** *metric-value*	Optional. By default, the default metric of the introduced routes of other protocols is 1.
Configure RIPng to redistribute routes.	**redistribute** *protocol* [*protocol-id*] [**metric** *metric-value*] [**route-map** *route-map-name*] [**match** *route-sub-type*]	Mandatory. By default, route redistribution is not configured.

Note:

- If the metric command option is specified during redistribution, the redistributed route adopts the metric.
- In configuring RIPng to redistribute routes for applying the route map, the available match options include ipv6 address, route type, tag, interface, ipv6 nexthop, ipv6 route-source, and metric, and the available set options include metric and tag.

5.2.3. Configure RIPng Route Control

Configuration Condition

Before configuring RIPng route control, ensure that:

- The IPv6 capability of the interface is enabled.
- RIPng is enabled.

Configure the Administrative Distance of RIPng

One device can run multiple routing protocols at the same time. The device selects the optimal route from the routes that are learnt from different protocols based on the administrative distances. The smaller the administrative distance is, the higher the priority is.

Table 5-5 Configure the administrative distance of RIPng

Step	**Command**	**Description**
Enter the global configuration mode.	**configure terminal**	-
Enter the RIPng configuration mode.	**ipv6 router rip** *process-id*	-
Configure the administrative distance of RIPng.	**distance** *distance-value*	Mandatory. By default, the administrative distance of RIPng is 120.

Configure a RIPng Route Summary

RIPng route summary always indicates configuring a pair of destination addresses and masks, which summarizes the routes in the covered network segment.

After RIP route summary is configured, the device advertises only the summary route. This greatly decreases the size of adjacent RIPng route tables in the medium and large networks, and decreases the consumption of the routing protocol packets for the network bandwidth.

The metric of the summary route adopts the minimum of all subnet route metrics.

Table 5-6 Configure the RIPng route summary function

Step	**Command**	**Description**
Enter the global configuration mode.	**configure terminal**	-
Enter the interface configuration mode.	**interface** *interface-name*	-

Step	Command	Description
Configure the route summary function of RIPng on the interface	**ipv6 rip summary-address** *prefix-address*	Mandatory By default, do not configure the route summary function.

Configure the RIPng Metric Offset

By default, RIPng adopts the route metric advertised by the neighbor device for the received routes. To modify the metric in some special application scenarios, you can configure the RIP metric offset to correct the metric of the specified route.

If the metric in the incoming direction is configured, RIPng modifies the metric of the received routes and saves the routes into the routing table. When RIPng advertises a metric to the neighbor devices, it advertises the new metric. If the metric in the outgoing direction is configured, the metric is modified only when RIPng advertises a metric to the neighbor devices.

Table 5-7 Configure the RIPng metric offset

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the RIPng configuration mode.	**ipv6 router rip** *process-id*	Mandatory By default, the RIPng process is disabled.
Configure RIPng to modify the metric of the specified route.	**offset-list** *access-list-name* { **in** | **out** } *metric-offset* [*interface-name*]	Mandatory By default, no metric is configured for any interface.

Configure RIPng Route Filtration

A router can filter the received or advertised routes by configuring an Access Control List (ACL) or prefix list. In receiving RIPng routes, you can filter some learnt routes; or in advertising RIPng routes, you can filter some routes that are advertised to neighbor devices.

Table 5-8 Configure RIPng route filtration

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the RIPng configuration mode.	**ipv6 router rip** *process-id*	Mandatory By default, the RIPng process is disabled.
Configure the RIPng route filtering function	**distribute-list** { *access-list-name* **| prefix** *prefix-list-name* **| route-map** *route-map-name*} { **in | out** } [*interface-name*]	Mandatory By default, the route filtering function is not configured. During the configuration process of the route filtering function, if no interface is specified, route filtering is enabled for all RIPng interfaces.

Configure the Metric of the RIPng Interface

After the interface enables RIPng, the corresponding direct route is generated in the database, with the default metric 1. When the route is in the RIPng database or it is advertised to neighbor devices, and if the metric is configured on the interface, adopt the interface metric.

If the interface metric is changed, the RIPng database immediately updates the corresponding direct route of RIPng and advertises the new metric to the neighbor devices.

Table 5-9 Configure the metric of the RIPng interface

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the interface configuration mode.	**interface** *interface-name*	-

Step	Command	Description
Configure the metric of the RIPng interface	**ipv6 rip metric** *metric-value*	Mandatory By default, the RIPng interface metric is 1.

Note:

- Configuring the RIPng interface metric affects only the metric of the direct subnet on the interface, while it does not affect the metric learned by the route.

Configure the Routing Tag for a RIPng Interface

The network administrator can attach tags to some routes. Then, when applying a routing policy, perform route filtering or route property advertisement based on the tags.

Table 5-10 Configure the routing flag for a RIPng interface

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the interface configuration mode.	**interface** *interface-name*	-
Configure a RIPng route tag of the direct subnet on the interface.	**ipv6 rip tag** *tag-value*	Mandatory By default, do not configure the route tag.

Configure the Maximum Number of RIPng Load Balancing Entries

This command helps you to control the number of the load balancing entries of the RIPng route.

Table 5-11 Configure the maximum number of RIPng load balancing entries

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-

Step	Command	Description
Enter the RIPng configuration mode.	**ipv6 router rip** *process-id*	Mandatory By default, the RIPng process is disabled.
Configure the maximum number of RIPng load balancing entries	**maximum-paths** *max-number*	Optional. By default, the maximum number of RIPng load balancing entries is 4.

5.2.4. Configure RIPng Network Optimization

Configuration Condition

Before configuring RIPng network optimization, ensure that:

- Configure the interface to enable the IPv6 capability
- Enable the RIPng protocol

Configure RIPng Timers

RIPng does not maintain neighbor relations and does not support route withdrawn; therefore, the protocol provides four configurable timers to control the network convergence speed. The four timers are: route update timer, router timeout timer, route dampening update timer, and route clear timer.

The route timeout time must be at least three times of the route update time. If no route update packet is received within the route timeout time, the route becomes invalid and it enters a dampening cycle. The length of the dampening cycle is determined by the dampening update time. During the cycle, the route will not be cleared. After the dampening cycle is completed, the route enters the clear cycle. During the cycle, the route can be updated. However, if no route update packet is received during the cycle, the route will be deleted.

Table 5-12 Configuring RIPng timers

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the RIPng configuration mode.	**ipv6 router rip** *process-id*	Mandatory By default, the RIPng process is disabled.

Step	Command	Description
Configure RIPng timers	**timers** *update-interval invalid-interval holddown-interval flush-interval*	Optional. By default, the RIPng update interval is 30s, the valid time for advertisement is 180s, the dampening time is 0s, and the clear time is 240s.

Note:

- In the same RIPng routing domain, the **timer** configurations on all the devices must be the same to prevent network flapping.

Configure RIPng Split Horizon and Toxicity Reverse of RIP

Split horizon and toxicity reverse are mechanisms that are used to prevent route loops.

1. Configure split horizon.

RIPng does not advertise routes that it has learnt from an interface to the interface, preventing routing loops.

Table 5-13 Configure RIPng split horizon

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the interface configuration mode.	**interface** *interface-name*	-
Configuring RIPng split horizon.	**no ipv6 rip split-horizon** [**disable**]	Optional By default, the split horizon function is enabled.

1. Configure toxicity reverse.

RIPng advertises the routes that have been learnt from an interface to the interface, but the route metric is the maximum number of hops 16, preventing routing loops.

Table 5-14 Configure RIPng toxicity reverse

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the interface configuration mode.	**interface** *interface-name*	-
Configure RIPng toxicity reverse	**ipv6 rip split-horizon poison-reverse**	Mandatory By default, the toxicity reverse function is disabled.

Note:

- The split horizon and toxicity reverse functions are valid only for the learnt routes, direct routes of the RIPng interface, and the redistributed direct and static routes.
- The split horizon function and the toxicity reversion function cannot be used at the same time.

Configure a Static RIPng Neighbor

RIPng does not maintain neighbor relations, so it does not have the concept of neighbor. Here the neighbor refers to the neighbor RIPng routing device. After a static RIPng neighbor is specified, RIPng sends RIPng packets to the neighbor in unicast mode. The configuration is applied to a network that does not support broadcast or multicast, such as point-to-point links. If the configuration is applied to a broadcast or multicast network, it may cause repeated RIPng packets in the network.

Table 5-15 Configure a static RIPng neighbor

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the interface configuration mode.	**interface** *interface-name*	-

Step	Command	Description
Configure advertisement of routes to a neighbor in unicast mode.	**ipv6 rip neighbor** *ipv6-address*	Mandatory The parameter ipv6-address is the IPv6 address of the peer direct-connect interface.

Note:

- RIPng advertises routes only to the interfaces that it covers, and **ipv6 rip passive** cannot prevent an interface from sending packets to its static neighbor.

Configure a Passive RIPng Interface

To decrease the network bandwidth consumed by the routing protocol, the dynamic routing protocol uses the passive interface function. RIPng receives only route update packets on a passive interface, and it does not send route update packets on the passive interface. In a low-speed network with small bandwidth, the passive interface function and the neighbor function cooperate to effectively reduce interactions of RIPng routes.

Table 5-16 Configure a passive RIPng interface

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the interface configuration mode.	**interface** *interface-name*	-
Configure a passive RIPng interface.	**ipv6 rip passive**	Mandatory By default, no passive interface is configured.

Note:

- **ipv6 rip passive** does not restrain an interface from sending unicast route updates to its neighbor devices. When being used with the **neighbor** command, **ipv6 rip passive** does not restrain an interface from sending unicast route updates to its neighbor devices. This application mode can control a router so that it sends route updates only to some neighbor devices in unicast mode instead of sending route updates to all neighbor devices in multicast mode.

5.2.5. RIPng Monitoring and Maintaining

Table 5-17 RIPng monitoring and maintaining

Command	Description
clear ipv6 rip [*process-id*]{ **process** | **statistics** }	Clears the RIPng process and statistics information
show ipv6 rip [*process-id*]	Displays the RIPng protocol basic information
show ipv6 rip [*process-id*] **database** [**detail** | *ipv6-address/mask-length* [**detail** | **longer-prefixes**]]	Displays the RIPng route database information
show ipv6 rip [*process-id*] **statistics** [*interface-name*]	Displays the RIPng interface statistics information
show ipv6 rip interface [*interface-name*]	Displays the RIPng interface information

5.3. RIPng Typical Configuration Example

5.3.1. Configure RIPng Basic Functions

Network Requirements

- Run RIPng between Device1 and Device2 for route interaction.

Network Topology

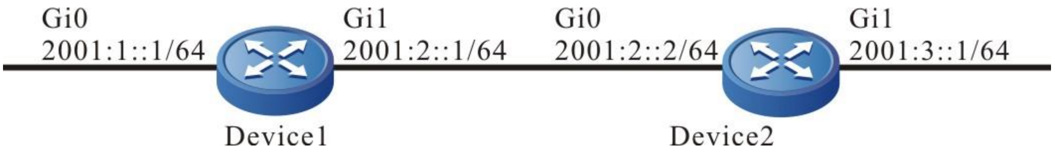

Figure 5-1 Networking for configuring RIPng basic functions

Configuration Steps

Step 1: Configure the IPv6 address of an interface (omitted).

Step 2: Configure RIPng.

#Configure Device1.

```
Device1#configure terminal
Device1(config)#ipv6 router rip 100
Device1(config-ripng)#exit
```

```
Device1(config)#interface gigabitethernet0
Device1(config-if-gigabitethernet0)#ipv6 rip enable 100
Device1(config-if-gigabitethernet0)#exit
Device1(config)#interface gigabitethernet1
Device1(config-if-gigabitethernet1)#ipv6 rip enable 100
Device1(config-if-gigabitethernet1)#exit
```

#Configure Device2.

```
Device2#configure terminal
Device2(config)#ipv6 router rip 100
Device2(config-ripng)#exit
Device2(config)#interface gigabitethernet0
Device2(config-if-gigabitethernet0)#ipv6 rip enable 100
Device2(config-if-gigabitethernet0)#exit
Device2(config)#interface gigabitethernet1
Device2(config-if-gigabitethernet1)#ipv6 rip enable 100
Device2(config-if-gigabitethernet1)#exit
```

Step 3: Check the result.

#View the IPv6 route table of Device1.

```
Device1#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management

L   ::1/128 [0/0]
    via ::, 2w4d:19:31:05, lo0
C   2001:1::/64 [0/0]
    via ::, 00:21:42, gigabitethernet0
L   2001:1::1/128 [0/0]
    via ::, 00:21:40, lo0
C   2001:2::/64 [0/0]
    via ::, 00:21:34, gigabitethernet1
L   2001:2::1/128 [0/0]
    via ::, 00:21:33, lo0
R   2001:3::/64 [120/2]
    via fe80::201:7aff:fec3:38a4, 00:11:19, gigabitethernet1
```

#Query the IPv6 routing table of Device2.

```
Device2#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management

L   ::1/128 [0/0]
    via ::, 3d:22:39:31, lo0
R   2001:1::/64 [120/2]
    via fe80::201:7aff:fe01:204, 00:12:00, gigabitethernet0
C   2001:2::/64 [0/0]
    via ::, 00:30:46, gigabitethernet0
L   2001:2::2/128 [0/0]
    via ::, 00:30:45, lo0
C   2001:3::/64 [0/0]
    via ::, 00:29:12, gigabitethernet1
L   2001:3::1/128 [0/0]
    via ::, 00:29:11, lo0
```

According to the routing table, you can see that the route advertised by the device uses a 64-bit exact mask.

5.3.2. Configure RIPng to Redistribute Routes

Network Requirements

- Run the IPv6 OSPF protocol between Device1 and Device2, Device2 learns the IPv6 OSPF route released by Device1 2001:1::/64, 2001:2::/64.
- Run the RIPng protocol between Device2 and Device3, Device2 only distributes the IPv6 OSPF route 2001:1::/64 to RIPng, and advertises the route to Device3.

Network Topology

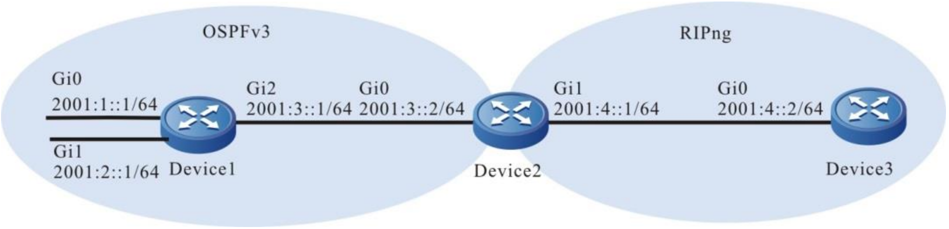

Figure 5-2 Networking for configuring RIPng to redistribute the route

Configuration Steps

Step 1: Configure the IPv6 address of an interface (omitted).

Step 2: Configure IPv6 OSPF.

#Configure Device1.

```
Device1#configure terminal
```

```
Device1(config)#ipv6 router ospf 100
Device1(config-ospf6)# router-id 1.1.1.1
Device1(config-ospf6)#exit
Device1(config)#interface gigabitethernet0
Device1(config-if-gigabitethernet0)#ipv6 router ospf tag 100 area 0
Device1(config-if-gigabitethernet0)#exit
Device1(config)#interface gigabitethernet1
Device1(config-if-gigabitethernet1)#ipv6 router ospf tag 100 area 0
Device1(config-if-gigabitethernet1)#exit
Device1(config)#interface gigabitethernet2
Device1(config-if-gigabitethernet2)#ipv6 router ospf tag 100 area 0
Device1(config-if-gigabitethernet2)#exit
```

#Configure Device2.

```
Device2#configure terminal
Device2(config)#ipv6 router ospf 100
Device2(config-ospf6)#router-id 2.2.2.2
Device2(config-ospf6)#exit
Device2(config)#interface gigabitethernet0
Device2(config-if-gigabitethernet0)#ipv6 router ospf tag 100 area 0
Device2(config-if-gigabitethernet0)#exit
```

#Query the IPv6 route table of Device2.

```
Device2#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management

L   ::1/128 [0/0]
    via ::, 4d:00:09:49, lo0
O   2001:1::/64 [110/2]
    via fe80::201:7aff:fe01:204, 00:12:16, gigabitethernet0
O   2001:2::/64 [110/2]
    via fe80::201:7aff:fe01:204, 00:12:16, gigabitethernet0
C   2001:3::/64 [0/0]
    via ::, 00:19:51, gigabitethernet0
L   2001:3::2/128 [0/0]
    via ::, 00:19:50, lo0
C   2001:4::/64 [0/0]
```

```
          via ::, 00:45:13, gigabitethernet1
        L 2001:4::1/128 [0/0]
          via ::, 00:45:12, lo0
```

According to the routing table, you can see that Device2 has learnt the IPv6 OSPF route advertised by Device1.

Step 3: Configure RIPng.

#Configure Device2.

```
Device2(config)#ipv6 router rip 100
Device2(config-ripng)#exit
Device2(config)#interface gigabitethernet1
Device2(config-if-gigabitethernet1)#ipv6 rip enable 100
Device2(config-if-gigabitethernet1)#exit
```

#Configure Device3.

```
Device3#configure terminal
Device3(config)#ipv6 router rip 100
Device3(config-ripng)#exit
Device3(config)#interface gigabitethernet0
Device3(config-if-gigabitethernet0)#ipv6 rip enable 100
Device3(config-if-gigabitethernet0)#exit
```

Step 4: Configure the routing policy.

#On Device2, configure route-map to invoke the prefix list to match 2001:1::/64 and filter 2001:2::/64.

```
Device2(config)#ipv6 prefix-list OSPF permit 2001:1::/64
Device2(config)#route-map OSPFtoRIP
Device2(config-route-map)#match ipv6 address prefix-list OSPF
Device2(config-route-map)#exit
```

Note:

- In configuring a routing policy, you can create a filtration rule based on a prefix list or ACL. The prefix list can precisely match routing masks while the ACL cannot match routing masks.

Step 5: Configure RIPng to redistribute IPv6 OSPF routes.

#Configure RIPng to redistribute IPv6 OSPF routes.

```
Device2(config)#ipv6 router rip 100
Device2(config-ripng)#redistribute ospf 100 route-map OSPFtoRIP
Device2(config-ripng)#exit
```

Step 6: Check the result.

#Query the RIPng database of Device2.

```
Device2#show ipv6 rip database
Type : N - Network interface, L - Learn, R - Redistribute, D - Default config,
     S - Static config
Proto: C - connected, S - static, R - RIP, O - OSPF, E - IRMP,
     o - SNSP, B - BGP, i-ISIS

RIPng process 100 routing database (VRF Kernel, Counter 2):
[Type/Proto]
[R/O] 2001:1::/64 metric 1
         via gigabitethernet0, fe80::201:7aff:fe01:204, no expires
[N/C] 2001:4::/64 metric 1, installed
         via gigabitethernet1, ::, no expires
```

#Query the IPv6 route table of Device3.

```
Device3#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
     U - Per-user Static route
     O - OSPF, OE-OSPF External, M - Management

L   ::1/128 [0/0]
    via ::, 2w0d:20:00:11, lo0
R   2001:1::/64 [120/2]
    via fe80::201:7aff:fec3:38a5, 02:50:14, gigabitethernet0
C   2001:4::/64 [0/0]
    via ::, 03:56:24, gigabitethernet0
L   2001:4::2/128 [0/0]
    via ::, 03:56:23, lo0
```

By querying the database of Device2 and the route table of Device3, it is found that the route on Device2 2001:1::/64 is re-distributed to RIPng and is successfully advertised to Device3, while the route 2001:2::/64 has been successfully filtered out.

Caution:

- In an actual application, if there are two or more AS boundary routers, it is recommended that you do not redistribute routes between different routing protocols. If route redistribution must be configured, you are required to configure route control policies such as route filtering and filtration summary on the AS boundary routers to prevent routing loops.

5.3.3. Configure RIPng Metric Offset

Network Requirements

- Device1, Device2, Device3, and Device4 runs the RIPng protocol and interconnects with each other.
- Device1 learns route 2001:5::/64 from both Device2 and Device3.
- On Device1, set the route metric offset in the receive direction so that Device1 selects the route advertised by Device2 with priority.

Network Topology

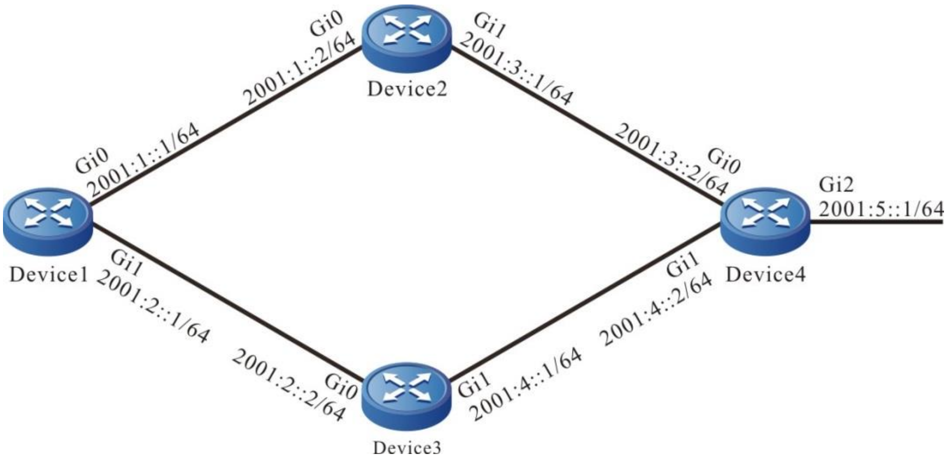

Figure 5-3 Networking for configuring the RIPng metric offset

Configuration Steps

Step 1: Configure the IPv6 address of an interface (omitted).

Step 2: Configure RIPng.

#Configure Device1.

```
Device1#configure terminal
Device1(config)#ipv6 router rip 100
Device1(config-ripng)#exit
Device1(config)#interface gigabitethernet0
Device1(config-if-gigabitethernet0)#ipv6 rip enable 100
Device1(config-if-gigabitethernet0)#exit
Device1(config)#interface gigabitethernet1
Device1(config-if-gigabitethernet1)#ipv6 rip enable 100
Device1(config-if-gigabitethernet1)#exit
```

#Configure Device2.

```
Device2#configure terminal
Device2(config)#ipv6 router rip 100
Device2(config-ripng)#exit
Device2(config)#interface gigabitethernet0
Device2(config-if-gigabitethernet0)#ipv6 rip enable 100
```

```
Device2(config-if-gigabitethernet0)#exit
Device2(config)#interface gigabitethernet1
Device2(config-if-gigabitethernet1)#ipv6 rip enable 100
Device2(config-if-gigabitethernet1)#exit
```

#Configure Device3.

```
Device3#configure terminal
Device3(config)#ipv6 router rip 100
Device3(config-ripng)#exit
Device3(config)#interface gigabitethernet0
Device3(config-if-gigabitethernet0)#ipv6 rip enable 100
Device3(config-if-gigabitethernet0)#exit
Device3(config)#interface gigabitethernet1
Device3(config-if-gigabitethernet1)#ipv6 rip enable 100
Device3(config-if-gigabitethernet1)#exit
```

#Configure Device4.

```
Device4#configure terminal
Device4(config)#ipv6 router rip 100
Device4(config-ripng)#exit
Device4(config)#interface gigabitetherne0
Device4(config-if-gigabitethernet0)#ipv6 rip enable 100
Device4(config-if-gigabitethernet0)#exit
Device4(config)#interface gigabitethernet1
Device4(config-if-gigabitethernet1)#ipv6 rip enable 100
Device4(config-if-gigabitethernet1)#exit
Device4(config)#interface gigabitethernet2
Device4(config-if-gigabitethernet2)#ipv6 rip enable 100
Device4(config-if-gigabitethernet2)#exit
```

#View the IPv6 route table of Device1.

```
Device1#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management

L   ::1/128 [0/0]
    via ::, 2w5d:06:21:24, lo0
C   2001:1::/64 [0/0]
    via ::, 00:02:05, gigabitethernet0
```

```
L   2001:1::1/128 [0/0]
    via ::, 00:02:04, lo0
C   2001:2::/64 [0/0]
    via ::, 00:02:02, gigabitethernet1
L   2001:2::1/128 [0/0]
    via ::, 00:02:01, lo0
R   2001:3::/64 [120/2]
    via fe80::201:7aff:fec3:38a4, 00:02:03, gigabitethernet0
R   2001:4::/64 [120/2]
    via fe80::201:7aff:fe11:2214, 00:00:48, gigabitethernet1
R   2001:5::/64 [120/3]
    via fe80::201:7aff:fec3:38a4, 00:02:03, gigabitethernet0
             [120/3]
    via fe80::201:7aff:fe11:2214, 00:00:48, gigabitethernet1
```

In the route table of Device1, you can see two routes to 2001:5::/64.

Step 3: Configure the access list.

```
Device1(config)#ipv6 access-list extended RIPng
Device1(config-v6-list)#permit 10 2001:5::/64 any
Device1(config-v6-list)#exit
```

Step 4: Configure a metric offset.

#On Device1, configure the metric offset list and increase the metric of the route that has been learnt from interface gigabitethernet1 and matches ACL to 3.

```
Device1(config)# ipv6 router rip 100
Device1(config-ripng)#offset-list RIPng in 3 gigabitethernet 1
Device1(config-ripng)#exit
```

Step 5: Check the result.

#View the IPv6 route table of Device1.

```
Device1#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management

L   ::1/128 [0/0]
    via ::, 2w5d:06:34:28, lo0
C   2001:1::/64 [0/0]
    via ::, 00:15:09, gigabitethernet0
```

```
L   2001:1::1/128 [0/0]
    via ::, 00:15:08, lo0
C   2001:2::/64 [0/0]
    via ::, 00:15:06, gigabitethernet1
L   2001:2::1/128 [0/0]
    via ::, 00:15:05, lo0
R   2001:3::/64 [120/2]
    via fe80::201:7aff:fec3:38a4, 00:03:10, gigabitethernet0
R   2001:4::/64 [120/2]
    via fe80::201:7aff:fe11:2214, 00:03:10, gigabitethernet1
R   2001:5::/64 [120/3]
    via fe80::201:7aff:fec3:38a4, 00:03:10, gigabitethernet0
```

According to the routing table of Device1, the next-hop output interface of route 2001:5::/64 is only gigabitethernet0, indicating that Device1 has selected the route advertised by Device2 with priority.

Note:

- The route metric offset list can be applied to all interfaces or a specified interface, and it can be used in the receiving or advertising direction.

5.3.4. Configure RIPng Route Filtration

Network Requirements

- Run RIPng between Device1 and Device2 for route interaction.
- Device1 has learnt the two routes 2001:2::/64 and 2001:3::/64 advertised by Device2, and then, the route 2001:3::/64 is filtered in the advertising direction of Device2.

Network Topology

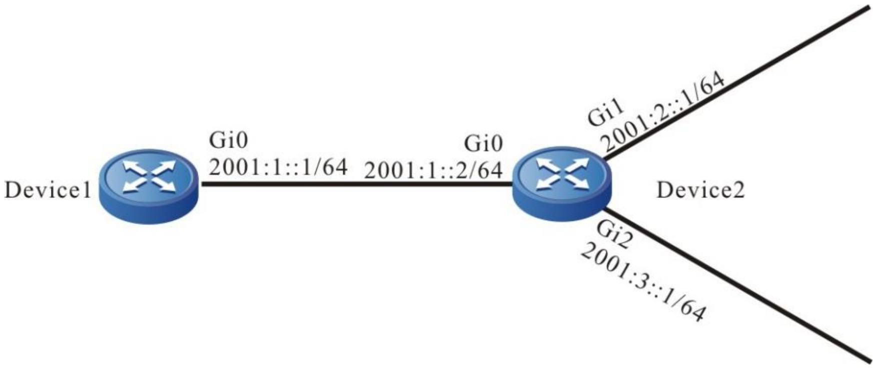

Figure 5-4 Networking for configuring RIPng Route filtration

Configuration Steps

Step 1: Configure the IPv6 address of an interface (omitted).

Step 2: Configure RIPng.

#Configure Device1.

```
Device1#configure terminal
```

```
Device1(config)#ipv6 router rip 100
Device1(config-ripng)#exit
Device1(config)#interface gigabitethernet0
Device1(config-if-gigabitethernet0)#ipv6 rip enable 100
Device1(config-if-gigabitethernet0)#exit
```

#Configure Device2.

```
Device2#configure terminal
Device2(config)#ipv6 router rip 100
Device2(config-ripng)#exit
Device2(config)#interface gigabitethernet0
Device2(config-if-gigabitethernet0)#ipv6 rip enable 100
Device2(config-if-gigabitethernet0)#exit
Device2(config)#interface gigabitethernet1
Device2(config-if-gigabitethernet1)#ipv6 rip enable 100
Device2(config-if-gigabitethernet1)#exit
Device2(config)#interface gigabitethernet2
Device2(config-if-gigabitethernet2)#ipv6 rip enable 100
Device2(config-if-gigabitethernet2)#exit
```

#View the IPv6 route table of Device1.

```
Device1#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management

L   ::1/128 [0/0]
    via ::, 2w5d:02:47:44, lo0
C   2001:1::/64 [0/0]
    via ::, 00:56:34, gigabitethernet0
L   2001:1::1/128 [0/0]
    via ::, 00:56:32, lo0
R   2001:2::/64 [120/2]
    via fe80::201:7aff:fec3:38a4, 00:27:11, gigabitethernet0
R   2001:3::/64 [120/2]
    via fe80::201:7aff:fec3:38a4, 00:27:11, gigabitethernet0
```

You can see that Device1 has learnt the two routes advertised by Device2.

Step 3: Configure the IPv6 prefix list.

```
Device2(config)#ipv6 prefix-list RIPng deny 2001:3::/64
```

Step 4: Configure the route filtration.

#Configure route filtering in the output direction of interface gigabitethernet0 of Device2.

```
Device2(config)#ipv6 router rip 100
Device2(config-ripng)#distribute-list prefix RIPng out gigabitethernet 0
Device2(config-ripng)#exit
```

Step 5: Check the result.

#View the IPv6 route table of Device1.

```
Device1#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
     U - Per-user Static route
     O - OSPF, OE-OSPF External, M - Management


L   ::1/128 [0/0]
    via ::, 2w5d:03:03:49, lo0
C   2001:1::/64 [0/0]
    via ::, 01:12:39, gigabitethernet0
L   2001:1::1/128 [0/0]
    via ::, 01:12:38, lo0
R   2001:2::/64 [120/2]
    via fe80::201:7aff:fec3:38a4, 00:43:16, gigabitethernet0
```

According to the routing table, Device2 does not advertise route 2001:3::/64 to Device1, but the route is deleted from the routing table of Device1 only after the route times out.

Note:

- The **distribute-list** can be applied to all interfaces or a specified interface, and it can be used in the receiving or advertising direction.

5.3.5. Configure RIPng Route Summary

Network Requirements

- Device1, Device2, Device3, and Device4 runs the RIPng protocol for the route interaction.
- Device1 has learnt two routes 2001:4:1:1::/64 and 2001:4:1:2::/64 from Device2. To reduce the size of the route table, Device2 needs to advertise the summary route of the two routes to Device1.

Network Topology

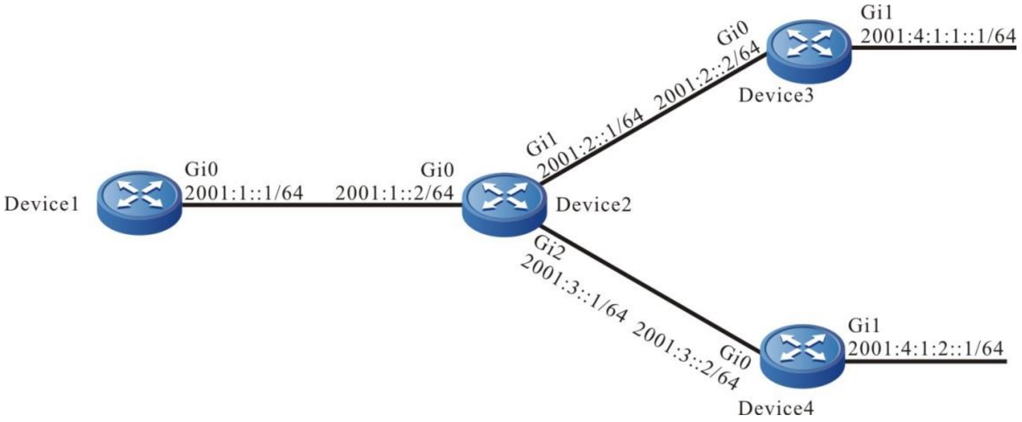

Figure 5-5 Networking for configuring RIPng route summary

Configuration Steps

Step 1: Configure the IPv6 address of an interface (omitted).

Step 2: Configure RIPng.

#Configure Device1.

```
Device1#configure terminal
Device1(config)#ipv6 router rip 100
Device1(config-ripng)#exit
Device1(config)#interface gigabitethernet0
Device1(config-if-gigabitethernet0)#ipv6 rip enable 100
Device1(config-if-gigabitethernet0)#exit
```

#Configure Device2.

```
Device2#configure terminal
Device2(config)#ipv6 router rip 100
Device2(config-ripng)#exit
Device2(config)#interface gigabitethernet0
Device2(config-if-gigabitethernet0)#ipv6 rip enable 100
Device2(config-if-gigabitethernet0)#exit
Device2(config)#interface gigabitethernet1
Device2(config-if-gigabitethernet1)#ipv6 rip enable 100
Device2(config-if-gigabitethernet1)#exit
Device2(config)#interface gigabitethernet2
Device2(config-if-gigabitethernet2)#ipv6 rip enable 100
Device2(config-if-gigabitethernet2)#exit
```

#Configure Device3.

```
Device3#configure terminal
```

```
Device3(config)#ipv6 router rip 100
Device3(config-ripng)#exit
Device3(config)#interface gigabitethernet0
Device3(config-if-gigabitethernet0)#ipv6 rip enable 100
Device3(config-if-gigabitethernet0)#exit
Device3(config)#interface gigabitethernet1
Device3(config-if-gigabitethernet1)#ipv6 rip enable 100
Device3(config-if-gigabitethernet1)#exit
```

#Configure Device4.

```
Device4#configure terminal
Device4(config)#ipv6 router rip 100
Device4(config-ripng)#exit
Device4(config)#interface gigabitethernet0
Device4(config-if-gigabitethernet0)#ipv6 rip enable 100
Device4(config-if-gigabitethernet00)#exit
Device4(config)#interface gigabitethernet1
Device4(config-if-gigabitethernet1)#ipv6 rip enable 100
Device4(config-if-gigabitethernet1)#exit
```

#View the IPv6 route table of Device1.

```
Device1#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
    U - Per-user Static route
    O - OSPF, OE-OSPF External, M - Management

L   ::1/128 [0/0]
   via ::, 2w5d:02:27:40, lo0
C   2001:1::/64 [0/0]
   via ::, 00:36:29, gigabitethernet0
L   2001:1::1/128 [0/0]
   via ::, 00:36:28, lo0
R   2001:2::/64 [120/2]
   via fe80::201:7aff:fec3:38a4, 00:07:06, gigabitethernet0
R   2001:3::/64 [120/2]
   via fe80::201:7aff:fec3:38a4, 00:07:06, gigabitethernet0
R   2001:4:1:1::/64 [120/3]
   via fe
80::201:7aff:fec3:38a4, 00:07:06, gigabitethernet0
```

```
R   2001:4:1:2::/64 [120/3]
    via fe80::201:7aff:fec3:38a4, 00:06:55, gigabitethernet0
```

Step 3: Configure the route summary of the interface.

#On Device2, configure the summary route 2001:4:1::/48.

```
Device2(config)#interface gigabitethernet0
Device2(config-if-gigabitethernet0)#ipv6 rip summary-address 2001:4:1::/48
Device2(config-if-gigabitethernet0)#exit
```

Step 4: Check the result.

#View the IPv6 route table of Device1.

```
Device1#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
      U - Per-user Static route
      O - OSPF, OE-OSPF External, M - Management

L   ::1/128 [0/0]
    via ::, 2w5d:02:35:44, lo0
C   2001:1::/64 [0/0]
    via ::, 00:44:33, gigabitethernet0
L   2001:1::1/128 [0/0]
    via ::, 00:44:32, lo0
R   2001:2::/64 [120/2]
    via fe80::201:7aff:fec3:38a4, 00:15:10, gigabitethernet0
R   2001:3::/64 [120/2]
    via fe80::201:7aff:fec3:38a4, 00:15:10, gigabitethernet0
R   2001:4:1::/48 [120/3]
    via fe80::201:7aff:fec3:38a4, 00:05:19, gigabitethernet0
```

You can see that Device1 has learnt the summary route 2001:4:1::/48 advertised by Device2, but the two detailed routes can be deleted from the route table only after timeout.

5.3.6. Configure Passive RIPng Interface

Network Requirements

- RIPng runs between Device1 and Device2 for route interaction.
- On Device1, configure a passive interface, which does not send update packets to Device2.

Network Topology

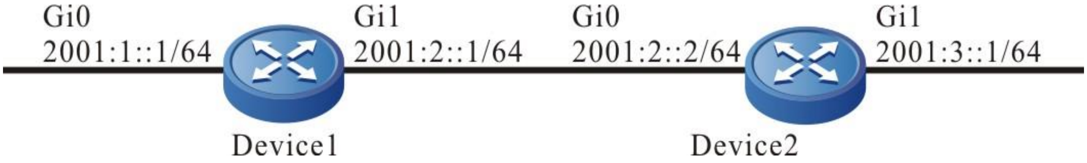

Figure 5-6 Networking for configuring an RIPng passive interface

Configuration Steps

Step 1: Configure the IPv6 address of an interface (omitted).

Step 2: Configure RIPng.

#Configure Device1.

```
Device1#configure terminal
Device1(config)#ipv6 router rip 100
Device1(config-ripng)#exit
Device1(config)#interface gigabitethernet0
Device1(config-if-gigabitethernet0)#ipv6 rip enable 100
Device1(config-if-gigabitethernet0)#exit
Device1(config)#interface gigabitethernet1
Device1(config-if-gigabitethernet1)#ipv6 rip enable 100
Device1(config-if-gigabitethernet1)#exit
```

#Configure Device2.

```
Device2#configure terminal
Device2(config)#ipv6 router rip 100
Device2(config-ripng)#exit
Device2(config)#interface gigabitethernet0
Device2(config-if-gigabitethernet0)#ipv6 rip enable 100
Device2(config-if-gigabitethernet0)#exit
Device2(config)#interface gigabitethernet1
Device2(config-if-gigabitethernet1)#ipv6 rip enable 100
Device2(config-if-gigabitethernet1)#exit
```

#View the IPv6 route table of Device1.

```
Device1#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
    U - Per-user Static route
    O - OSPF, OE-OSPF External, M - Management

L   ::1/128 [0/0]
   via ::, 2w4d:19:31:05, lo0
```

```
C   2001:1::/64 [0/0]
    via ::, 00:21:42, gigabitethernet0
L   2001:1::1/128 [0/0]
    via ::, 00:21:40, lo0
C   2001:2::/64 [0/0]
    via ::, 00:21:34, gigabitethernet1
L   2001:2::1/128 [0/0]
    via ::, 00:21:33, lo0
R   2001:3::/64 [120/2]
    via fe80::201:7aff:fec3:38a4, 00:11:19, gigabitethernet1
```

#Query the IPv6 route table of Device2.

```
Device2#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management

L   ::1/128 [0/0]
    via ::, 3d:22:39:31, lo0
R   2001:1::/64 [120/2]
    via fe80::201:7aff:fe01:204, 00:12:00, gigabitethernet0
C   2001:2::/64 [0/0]
    via ::, 00:30:46, gigabitethernet0
L   2001:2::2/128 [0/0]
    via ::, 00:30:45, lo0
C   2001:3::/64 [0/0]
    via ::, 00:29:12, gigabitethernet1
L   2001:3::1/128 [0/0]
    via ::, 00:29:11, lo0
```

Step 3: Configure a passive interface.

#Configure Device1.

```
Device1(config)#interface gigabitethernet1
Device1(config-if-gigabitethernet1)#ipv6 rip passive
Device1(config-if-gigabitethernet1)#exit
```

gigabitethernet1 of Device1 is configured as a passive interface, which does not send update packets to Device2, but still can receive update packets.

Step 4: Check the result.

#View the IPv6 route table of Device1.

```
Device1#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management

L   ::1/128 [0/0]
    via ::, 2w4d:19:55:37, lo0
C   2001:1::/64 [0/0]
    via ::, 00:46:14, gigabitethernet0
L   2001:1::1/128 [0/0]
    via ::, 00:46:12, lo0
C   2001:2::/64 [0/0]
    via ::, 00:46:06, gigabitethernet1
L   2001:2::1/128 [0/0]
    via ::, 00:46:05, lo0
R   2001:3::/64 [120/2]
    via fe80::201:7aff:fec3:38a4, 00:35:51, gigabitethernet1
```

Route 2001:3::/64 is still kept on Device1. On Device2, after the RIPng route times out and is deleted, route 2001:1::/64 is deleted from the routing table.

#Query the IPv6 route table of Device2.

```
Device2#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management

L   ::1/128 [0/0]
    via ::, 3d:23:05:24, lo0
C   2001:2::/64 [0/0]
    via ::, 00:56:39, gigabitethernet0
L   2001:2::2/128 [0/0]
    via ::, 00:56:38, lo0
C   2001:3::/64 [0/0]
    via ::, 00:55:05, gigabitethernet1
L   2001:3::1/128 [0/0]
    via ::, 00:55:04, lo0
```

5.3.7. Configure RIPng to Use IPSec Encryption Authentication

Network Requirements

- Run RIPng between Device1 and Device2.
- Device1 and Device2 use the IPSec tunnel to perform encryption authentication for the RIPng packets.
- After configuration, the device can perform routing interaction normally.

Network Topology

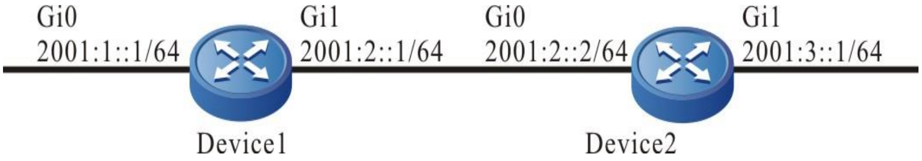

Figure 5-7 Networking of configuring the RIPng authentication function

Configuration Steps

Step 1: Configure the IPv6 address of the interface (omitted).

Step 2: Configure the RIPng process and enable the RIPng function on the interface.

#Configure Device1.

```
Device1#configure terminal
Device1(config)#ipv6 router rip 100
Device1(config-ripng)#exit
Device1(config)#interface gigabitethernet0
Device1(config-if-gigabitethernet0)#ipv6 rip enable 100
Device1(config-if-gigabitethernet0)#exit
Device1(config)#interface gigabitethernet1
Device1(config-if-gigabitethernet1)#ipv6 rip enable 100
Device1(config-if-gigabitethernet1)#exit
```

#Configure Device2.

```
Device2#configure terminal
Device2(config)#ipv6 router rip 100
Device2(config-ripng)#exit
Device2(config)#interface gigabitethernet0
Device2(config-if-gigabitethernet0)#ipv6 rip enable 100
Device2(config-if-gigabitethernet0)#exit
Device2(config)#interface gigabitethernet1
Device2(config-if-gigabitethernet1)#ipv6 rip enable 100
Device2(config-if-gigabitethernet1)#exit
```

Step 3: Configure the IPSec proposal and manual tunnel.

#Configure Device1, create IPSec proposal a, adopt ESP transmission encapsulation mode, encryption algorithm 3DES, authentication algorithm sha1, create IPSec manual tunnel a, and configure SPI and key.

```
Device1(config)#crypto ipsec proposal a
Device1(config-ipsec-prop)#mode transport
Device1(config-ipsec-prop)#esp 3des sha1
Device1(config-ipsec-prop)#exit
Device1(config)#crypto ipv6-tunnel a manual
Device1(config-manual-tunnel)#set ipsec proposal a
Device1(config-manual-tunnel)#set inbound esp 1000 encryption 0 111111111111111111111111 authentication 0 aaaaaaaaaaaaaaaaaaaa
Device1(config-manual-tunnel)#set outbound esp 1001 encryption 0 aaaaaaaaaaaaaaaaaaaaaaaa authentication 0 11111111111111111111
Device1(config-manual-tunnel)#exit
```

#Configure Device2, create IPSec proposal a, adopt ESP transmission encapsulation mode, encryption algorithm 3des, authentication algorithm sha1, create IPSec manual tunnel a, and configure SPI and key.

```
Device2(config)#crypto ipsec proposal a
Device2(config-ipsec-prop)#mode transport
Device2(config-ipsec-prop)#esp 3des sha1
Device2(config-ipsec-prop)#exit
Device2(config)#crypto ipv6-tunnel a manual
Device2(config-manual-tunnel)#set ipsec proposal a
Device2(config-manual-tunnel)#set inbound esp 1001 encryption 0 aaaaaaaaaaaaaaaaaaaaaaaa authentication 0 11111111111111111111
Device2(config-manual-tunnel)#set outbound esp 1000 encryption 0 111111111111111111111111 authentication 0 aaaaaaaaaaaaaaaaaaaa
Device2(config-manual-tunnel)#exit
```

Step 4: In the RIPng process, the interface is bound with the corresponding IPSec tunnel.

#On the interface Gigabitethernet1 of Device1, bind the IPSec tunnel a.

```
Device1(config)#interface gigabitethernet1
Device1(config-if-gigabitethernet1)#ipv6 rip ipsec-tunnel a
Device1(config-if-gigabitethernet1)#exit
```

#On the interface Gigabitethernet0 of Device2, bind the IPSec tunnel a.

```
Device2(config)#interface gigabitethernet0
Device2(config-if-gigabitethernet0)#ipv6 rip ipsec-tunnel a
Device2(config-if-gigabitethernet0)#exit
```

Step 5: Check the result.

#View the RIPng interface information of Device1 Gigabitethernet1.

```
Device1#show ipv6 rip interface gigabitethernet 1
gigabitethernet1 is up, line protocol is up
  RIPng enable status       : Enable on process 100
  RIPng running status      : Up
  VPN Routing/Forwarding    : Kernel
  Passive interface         : Disabled
  Split horizon             : enable
  Packet MTU                : 1500
  Joined RIPng multicast    : Yes
  IPv6 interface address    :
    2001:2::1/64
    fe80::201:7aff:fe74:55e7/10
  RIPng bfd interface       : Enable
  RIPng bfd function open   : OFF
  RIPng bfd interface state : DOWN
  RIPng ipsec-tunnel info:
    Bind to ipsec-tunnel :a
    Tunnel-id: 0
```

#View the IPv6 route table of Device1.

```
Device1#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management

L   ::1/128 [0/0]
     via ::, 2w4d:19:31:05, lo0
C   2001:1::/64 [0/0]
     via ::, 00:21:42, gigabitethernet0
L   2001:1::1/128 [0/0]
     via ::, 00:21:40, gigabitethernet0
C   2001:2::/64 [0/0]
     via ::, 00:21:34, gigabitethernet1
L   2001:2::1/128 [0/0]
     via ::, 00:21:33, gigabitethernet1
```

```
R   2001:3::/64 [120/2]
    via fe80::201:7aff:fec3:38a4, 00:11:19, gigabitethernet1
```

#View the IPv6 route table of Device2.

```
Device2#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
     U - Per-user Static route
     O - OSPF, OE-OSPF External, M - Management

L   ::1/128 [0/0]
    via ::, 3d:22:39:31, lo0
R   2001:1::/64 [120/2]
    via fe80::201:7aff:fe01:204, 00:12:00, gigabitethernet0
C   2001:2::/64 [0/0]
    via ::, 00:30:46, gigabitethernet0
L   2001:2::2/128 [0/0]
    via ::, 00:30:45, gigabitethernet0
C   2001:3::/64 [0/0]
    via ::, 00:29:12, gigabitethernet1
L   2001:3::1/128 [0/0]
    via ::, 00:29:11, gigabitethernet1
```

It can be seen from the routing table that the IPSec tunnel has been bound successfully, the packet sending and receiving are normal, and the route can be learned from each other.

Note:

- When configuring RIPng to bind the IPSec tunnel, you can only configure the process binding or the interface binding, and you also can configure the process and interface binding at the same time.
- When the process binding and interface binding are configured for the IPSec tunnel at the same time, the interface binding takes effect first.

6. OSPF

6.1. Overview

Open Shortest Path First (OSPF) is a dynamic routing protocol that is based on link statuses. It uses the Dijkstra's Shortest Path First (SPF) algorithm to calculate routes within a single Autonomous System (AS).

OSPF, which is developed by the Internet Engineering Task Force (IETF), solves the problems of slow convergence and liability to form loops for distance vector routes. It is applicable to medium- and large-sized networks. Currently, OSPF version 2 is available. It complies with RFC2328 and supports OSPF extended functions defined in other related RFCs.

In OSPF, each device maintains a database that describes the link status of an AS network. The databases of devices in the same area are the same. After the databases are completely synchronized, each device takes itself as the root and uses the SPF algorithm to calculate the shortest path tree without loops to describe the shortest paths it knows to reach each destination. Then each device constructs its route table based on the shortest path tree.

The main features of OSPF include:

- Fast convergence: After the topology of the network changes, it sends an update packet immediately so that the change is synchronized in the AS.
- Loop free: OSPF runs SPF to calculate routes based on the link status database. The algorithm ensures that no routing loop will be formed.
- Dividing areas: OSPF allows to divide an AS into multiple areas to reduce network bandwidth occupancy, making it possible to construct layered network.
- Authentication support: Once an OSPF device receives a routing protocol packet, it verifies the authentication information contained in the packet to prevent information leakage or malicious attacks in the network.
- Supports subnet with different lengths: The routes advertised by OSPF carry network masks to support subnets with different lengths.
- Support load balancing: OSPF supports multiple equivalent routes to the same destination.

6.2. OSPF Function Configuration

Table 6-1 OSPF function list

Configuration Tasks	
Configure basic OSPF functions.	Enable OSPF.
Configure OSPF areas.	Configure an OSPF NSSA area.
	Configure an OSPF Stub area.
	Configure an OSPF virtual link.

Configuration Tasks	
Configure the OSPF network type.	Configure the network type of an OSPF interface to broadcast.
	Configure the network type of an OSPF interface to P2P.
	Configure the network type of an OSPF interface to NBMA.
	Configure the network type of an OSPF interface to P2MP.
Configure the OSPF network authentication.	Configure OSPF area authentication.
	Configure OSPF interface authentication.
Configure OSPF route generation.	Configure OSPF to redistribute routes.
	Configure the default OSPF route.
	Configure the OSPF host route.
Configure OSPF route control.	Configure route summary on inter-area OSPF routes.
	Configure OSPF external route summary.
	Configure route filtering on inter-area OSPF routes.
	Configure OSPF external route filtration.
	Configure OSPF route installation filtration.
	Configure the cost value of an OSPF interface.
	Configure the OSPF reference bandwidth.

Configuration Tasks	
Configure OSPF route control.	Configure the OSPF administrative distance.
	Configure the maximum number of OSPF load balancing routes.
	Configure OSPF to be compatible with RFC1583.
Configure OSPF network optimization.	Configure the keep-alive time of an OSPF neighbor.
	Configure an OSPF passive interface.
	Configure an OSPF demand circuit.
	Configure the priority of an OSPF interface.
	Configure the MTU of an OSPF interface.
	Configure the LSA transmit delay of an OSPF interface.
	Configure OSPF LSA retransmission.
	Configure OSPF LSA sending
	Configure OSPF to prevent LSA flooding.
	Configure OSPF SPF calculation time.
	Configure OSPF database overflow.
Configure OSPF to coordinate with BFD.	Configure OSPF to coordinate with BFD.
Configure the OSPF fast re-routing	Configure the OSPF fast re-routing

6.2.1. Configure Basic OSPF Functions

Before configuring OSPF functions, you must first enable the OSPF protocol before the other functions can take effect.

Configuration Conditions

Before configuring the basic OSPF functions, ensure that:

- The link layer protocol has been configured to ensure normal communication at the link layer.
- Interface IP addresses have been configured so that neighbor nodes are reachable at the network layer.

Enable OSPF

To enable OSPF, you must create an OSPF process, specify the address range of the networks with which the process is associated, and specify the area to which the address range belongs. If the IP address of an interface is in the network segment of an area, the interface belongs to the area and the OSPF function is enabled, and OSPF advertises the direct route of the interface.

A device that runs OSPF must have a Router ID, which is used to uniquely identify a device in an OSPF AS. You must ensure that the Router IDs are unique in an AS; otherwise, setup of neighbors and route learning are affected. A Router ID can be specified when the OSPF process is created. If the Router ID is not specified, it can be elected according to the following rules:

- Select the biggest IP address from loopback interface IP addresses as the Router ID.
- If no loopback interface is configured with an IP address, select the biggest IP address from the IP addresses of other interfaces as the Router ID.
- Only when an interface is in the UP status can the IP address of the interface be elected as the Router ID.

OSPF supports multiple processes, which are identified by different process numbers. The processes are independent of each other, and they do not affect each other.

Table 6-2 Enable OSPF

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Create an OSPF process and enter the OSPF configuration mode.	**router ospf** *process-id* [**vrf** *vrf-name*]	Mandatory. Enable the OSPF process or enable the OSPF process from VRF. By default, OSPF is disabled. If you enable OSPF from VRF, the OSPF process that belongs to a VRF can manage only interfaces under the VRF.

Step	Command	Description
Configure network segments that are covered by an OSPF area.	**network** *ip-address wildcard-mask* **area** *area-id*	Mandatory. By default, an interface does not belong to any OSPF process or area. A interface can only belong to an OSPF process and area.
Configure the Router ID of the OSPF process.	**router-id** *ip-address*	Optional. By default, the election rule based on Router ID is generated. Modifying Router ID will not make OSPF neighbor become invalid. To make the new Router ID take effect, you need to reset the process manually.

6.2.2. Configure OSPF Areas

To prevent a large amount of database information from occupying too much CPU and memory, you can divide an OSPF AS into multiple areas. An area can be identified with a 32-bit area ID, a decimal number in the range of 0-4294967295, or an IP address in the range of 0.0.0.0-255.255.255.255. Area 0 or 0.0.0.0 represents an OSPF backbone area, while other non-zero areas are non-backbone areas. All routing information between areas must be forwarded through the backbone area. Non-backbone areas cannot directly exchange routing information.

OSPF defines several types of routers:

- Internal router: All interfaces belong to the devices in one area.
- Area Border Router (ABR): It is connected to devices from different areas.
- Autonomous System Boundary Router (ASBR): It is a device that introduces external routes to the OSPF AS.

Configuration Condition

Before configuring an OSPF area, ensure that:

- Interface IP addresses have been configured so that neighbor nodes are reachable at the network layer.
- Enable OSPF.

Configure an OSPF NSSA Area

A Not-So-Stub-Area (NSSA) does not allow injection of Type-5 Link State Advertisement (LSA) but it allows injection of Type-7 LSA. External routes can be introduced to an NSSA area through redistribution of configuration. The ASBR in the NSSA area generate Type-7 LSAs and flood LSAs to the NSSA area. The ABR in an NSSA area converts Type-7 LSAs into Type-5 LSAs, and floods the converted Type-5 LSAs into the entire AS.

The OSPF NSSA area that is configured by using the **area** *area-id* **nssa no-summary** command is called a totally NSSA area. An OSPF totally NSSA area does not allow cross-area routes to flood in the area. At this time, the ABR generates a default route and flood it into the NSSA area. The devices in the NSSA area access a network outside the area through the default route.

Table 6-3 Configure an OSPF NSSA area

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the OSPF configuration mode.	**router ospf** *process-id* [**vrf** *vrf-name*]	-
Configure an NSSA area.	**area** *area-id* **nssa** [[**default-information-originate** [**metric** *metric-value* / **metric-type** *type-value*] / **no-redistribution** / **no-summary** / **translator-role** { **always** | **candidate** | **never** }] | [**translate-always** | **translate-candidate** | **translate-never**]]	Mandatory. By default, an area is not an NSSA area.

Note:

- A backbone area cannot be configured as an NSSA area.
- All devices in one NSSA area must be configured as NSSA areas, because devices with different area types cannot form neighbor relations.

Configure an OSPF Stub Area

A Stub area does not allow external route outside an AS to flood in the area so as to reduce the size of the link status database. After an area is configured as a Stub area, the ABR which is located at the Stub border generates a default route and flood the route into the Stub area. The devices in the Stub area access a network outside the area through the default route.

The OSPF Stub area that is configured by using the **area** *area-id* **stub no-summary** command is called a totally Stub area. An OSPF totally Stub area does not allow inter-area routes and external routes to flood in the area. The devices in the area access a network outside the area and outside the OSPF AS through the default route.

Table 6-4 Configure an OSPF stub area

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-

Step	Command	Description
Enter the OSPF configuration mode.	**router ospf** *process-id* [**vrf** *vrf-name*]	-
Configure a Stub area.	**area** *area-id* **stub** [**no-summary**]	Mandatory. By default, an area is not a Stub area.
Configure the ABR in the Stub area to generate the cost value of the default route.	**area** *area-id* **default-cost** *cost-value*	Optional. By default, the ABR of the Stub area sets the cost value of the default route to 1.

Note:

- A backbone area cannot be configured as a Stub area.
- All devices in one Stub area must be configured as Stub areas, because devices with different area types cannot form neighbor relations.

Configure an OSPF Virtual Link

The non-backbone areas in OSPF must synchronize and exchange data through the backbone area. Therefore, all non-backbone areas must keep connected with the backbone area.

If the requirement fails to be meet in certain cases, you can solve the problem by configuring a virtual link. After configuring a virtual link, you can configure an authentication mode for the virtual link and modify the Hello interval. The meanings of the parameters are the same as the meanings of the parameter of common OSPF interfaces.

Table 6-5 Configure an OSPF virtual link

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the OSPF configuration mode.	**router ospf** *process-id* [**vrf** *vrf-name*]	-

Step	Command	Description
Configure an OSPF virtual link.	**area** *transit-area-id* **virtual-link** *neighbor-id* [[**authentication** [**message-digest | null**] | **authentication-key** *key* | **message-digest-key** *key-id* { **md5 | sm3** } *key*] / **dead-interval** *seconds* **hello-interval** *seconds* / **retransmit-interval** *seconds* / **transmit-delay** *seconds*]	Mandatory. By default, no virtual link is created.

Note:

- A virtual link must be configured between two ABRs.
- Two ABRs on which the virtual link is configured must be in the same public area. This area is also called the transit area of the virtual link.
- The transit area of a virtual link must not be a Stub area or NSSA area.

6.2.3. Configure OSPF Network Type

According to the link protocol types, OSPF classifies networks into four types:

- Broadcast Network: When the link protocol of the network is Ethernet or Fiber Distributed Data Interface (FDDI), the default OSPF network type is broadcast network.
- Point To Point Network (P2P Network): When the link protocol is Point to Point Protocol (PPP), Link Access Procedure Balanced (LAPB), or High-level Data Link Control (HDLC), the default OSPF network type is P2P network.
- Non-Broadcast Multi-Access Network (NBMA Network): When the link protocol is ATM, frame relay, or X.25, the default OSPF network type is NBMA.
- Point To Multi-Point Network (P2MP): No link protocol will be regarded by OSPF as the P2MP network by default. Usually, the NBMA network that is not totally connected is configured as the OSPF P2MP network.

You can modify the network type of an OSPF interface according to the actual requirement. The network types of the interfaces through which OSPF neighbors are set up must be the same; otherwise, normal learning of routes is affected.

Configuration Condition

Before configuring the OSPF network type, ensure that:

- Interface IP addresses have been configured so that neighbor nodes are reachable at the network layer.
- Enable OSPF.

Configure the Network Type of an OSPF Interface to Broadcast

A broadcast network supports multiple devices (more than two devices). These devices can exchange information with all the devices in the network. OSPF uses Hello packets to dynamically discover neighbors.

Table 6-6 Configure the network type of an OSPF interface to broadcast

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the interface configuration mode.	**interface** *interface-name*	-
Configure the network type of an OSPF interface to broadcast.	**ip ospf network broadcast**	Mandatory. By default, the network type of an OSPF interface is determined by the link layer protocol.

Configure the Network Type of an OSPF Interface to P2P

A P2P network is a network that consists of two devices. Each device is located at one end of a P2P link. OSPF uses Hello packets to dynamically discover neighbors.

Table 6-7 Configure the network type of an OSPF interface to P2P

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the interface configuration mode.	**interface** *interface-name*	-
Configure the OSPF network type to P2P.	**ip ospf network point-to-point**	Mandatory. By default, the network type of an OSPF interface is determined by the link layer protocol.

Configure the Network Type of an OSPF Interface to NBMA

An NBMA network supports multiple devices (more than two devices), but the devices does not have the broadcast capability, therefore, you must specify a neighbor manually.

Table 6-8 Configure the network type of an OSPF interface to NBMA

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the interface configuration mode.	**interface** *interface-name*	-
Configure the OSPF network type to NBMA.	**ip ospf network non-broadcast**	Mandatory. By default, the network type of an OSPF interface is determined by the link layer protocol.
Enter the global configuration mode.	**exit**	-
Enter the OSPF configuration mode.	**router ospf** *process-id* [**vrf** *vrf-name*]	-
Configure a neighbor for the NBMA network.	**neighbor** *neighbor-ip-address* [**cost** *cost-value* / **priority** *priority-value* / **poll-interval** *interval-value*]	Mandatory. In an NBMA network, a neighbor must be specified manually.

Configure the Network Type of an OSPF Interface to P2MP

When an NBMA network is not fully connected, you can configure its network type to P2MP to save network overhead. If the network type is configured to P2MP unicast, you need to specify a neighbor manually.

Table 6-9 Configure the network type of an OSPF interface to P2MP

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the interface configuration mode.	**interface** *interface-name*	-

Step	Command	Description
Configure the OSPF network type to P2MP.	**ip ospf network point-to-multipoint** [**non-broadcast**]	Mandatory. By default, the network type of an OSPF interface is determined by the link layer protocol.
Enter the global configuration mode.	**exit**	-
Enter the OSPF configuration mode.	**router ospf** *process-id* [**vrf** *vrf-name*]	-
Configure a neighbor for the P2MP unicast network.	**neighbor** *neighbor-ip-address* [**cost** *cost-value* / **priority** *priority-value* / **poll-interval** *interval-value*]	If the interface network type is set to P2MP unicast, it is mandatory.

6.2.4. Configure OSPF Network Authentication

To prevent information leakage or malicious attacks to OSPF devices, all packet interaction between OSPF neighbors has the authentication capability. The authentication types include: NULL (no authentication), plain text authentication, MD5 authentication, and key-chain authentication.

If authentication is configured, an OSPF interface requires authentication before receiving OSPF protocol packets. The OSPF interface receives only packets that have passed authentication. Therefore, the OSPF interfaces through which neighbor relations are set up, their authentication modes, Key IDs, and authentication passwords must be the same.

An authentication mode and an authentication password are configured independently. If an authentication password has been configured but no authentication mode is configured, the authentication mode corresponding to the authentication password will be automatically configured.

An OSPF authentication mode can be configured on an area, interface, or interface address. The priorities that are sorted from low to high include: area authentication, interface authentication, and interface address authentication. That is, the interface address authentication is first used, and then the interface authentication, and finally the area authentication.

Configuration Condition

Before configuring OSPF authentication, ensure that:

- Interface IP addresses have been configured so that neighbor nodes are reachable at the network layer.
- Enable OSPF.

Configure OSPF Area Authentication

To validate OSPF area authentication, you must configure not only the area authentication mode but also the corresponding authentication password on the interface.

Table 6-10 Configure OSPF area authentication

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the OSPF configuration mode.	**router ospf** *process-id* [**vrf** *vrf-name*]	-
Configure the area authentication mode.	**area** *area-id* **authentication** [**message-digest** | key-chain]	Mandatory. By default, area authentication is not configured. The keyword **message-digest** in the command indicates MD5 authentication and the key word **key-chain** indicates the key-chain authentication. Otherwise, plain text authentication is configured.
Enter the interface configuration mode.	**interface** *interface-name*	-
Configure a password for plain text authentication.	**ip ospf** [*ip-address*] **authentication-key** { **0 | 7** } *password*	Mandatory. By default, no password is configured for plain text authentication.
Configure a password for MD5 or SM3 authentication.	**ip ospf** [*ip-address*] **message-digest-key** *key-id* { **md5 | sm3** } { **0** | **7** } *password*	Mandatory. By default, no password is configured for MD5 or SM3 authentication.

Step	Command	Description
Configure the key-chain authentication	**ip ospf** [*ip-address*] **key-chain** *key-chain name*	Mandatory By default, do not configure the key-chain authentication.

Configure OSPF Interface Authentication

If an OSPF interface has multiple IP addresses, you can set an authentication mode or authentication password for one IP address of the interface. If you do not specify an interface address, all addresses of the interface use the specified authentication mode or authentication password.

Table 6-11 Configure OSPF interface authentication

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the interface configuration mode.	**interface** *interface-name*	-
Configure the interface authentication mode.	**ip ospf** [*ip-address*] **authentication** [**key-chain | message-digest | null**]	Mandatory. By default, interface authentication mode is not configured. The keyword **message-digest** in the command indicates MD5 authentication, the key word **key-chain** indicates the key-chain authentication, and the keyword **null** indicates no authentication; otherwise, plain text authentication is configured.

Step	**Command**	**Description**
Configure a password for plain text authentication.	**ip ospf** [*ip-address*] **authentication-key** { **0 | 7** } *password*	Mandatory. By default, no password is configured for plain text authentication.
Configure a password for MD5 or SM3 authentication.	**ip ospf** [*ip-address*] **message-digest-key** *key-id* { **md5 | sm3** } { **0** | **7** } *password*	Mandatory. By default, no password is configured for MD5 or SM3 authentication.
Configure the key-chain authentication	**ip ospf** [*ip-address*] **key-chain** *key-chain name*	Mandatory By default, do not configure the key-chain authentication.

6.2.5. Configure OSPF Route Generation

OSPF uses the **network** command to cover routes of the directly connected network segment. It can also redistribute external routes or use the **host** command to add host routes.

Configuration Condition

Before configuring OSPF route generation, ensure that:

- Interface IP addresses have been configured so that neighbor nodes are reachable at the network layer.
- Enable OSPF.

Configure OSPF to Redistribute Routes

If multiple routing protocols run on one device, routes of other protocols can be introduced to OSPF through redistribution. By default, class 2 external routes of OSPF are generated, with the routing metric 20. When you introduces external routes through redistribution, you can modify the external route type, metric, and tag field, and configure the required routing policy to perform route control and management.

Table 6-12 Configure OSPF to redistribute routes

Step	**Command**	**Description**
Enter the global configuration mode.	**configure terminal**	-
Enter the OSPF configuration mode.	**router ospf** *process-id* [**vrf** *vrf-name*]	-

Step	Command	Description
Configure OSPF to redistribute routes.	**redistribute** *protocol* [*protocol-id*] [**metric** *metric-value* / **metric-type** *metric-type* / **tag** *tag-value* / **route-map** *route-map-name* / **match** *route-type*]	Mandatory. By default, route redistribution is not configured for OSPF.
Configure the metric of the OSPF external routes.	**default-metric** *metric-value*	Optional.

Note:

- If the metric value of external routes are configured by using both the **redistribute** *protocol* [*protocol-id*] **metric** command and the **default-metric** command, the value that is configured by using the former command has a higher priority.

Configure the Default OSPF Route

After an OSPF Stub area or a totally NSSA areas is configured, a Type-3 default route is generated. For an NSSA area, no default route is automatically generated. You can use the **area** *area-id* **nssa default-information-originate** command to introduce a Type-7 default route to the NSSA area.

OSPF cannot use the **redistribute** command to introduce a Type-5 default route. To do this, use the **default-information originate** [**always**] command.

Table 6-13 Configure the default OSPF route

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the OSPF configuration mode.	**router ospf** *process-id* [**vrf** *vrf-name*]	-

Step	Command	Description
Configure OSPF to introduce a default route.	**default-information originate** [**always** / **metric** *metric-value* / **metric-type** *metric-type* / **route-map** *route-map-name*]	Mandatory. By default, no external default route is introduced to an OSPF AS. The default metric of the introduced default route is 1, and the type is external type 2. The field **always** means to force the OSPF AS to generate a default route; otherwise, the default route is generated only when there is a default route in the local route table.

Configure the OSPF Host Route

Table 6-14 Configure the OSPF host route

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the OSPF configuration mode.	**router ospf** *process-id* [**vrf** *vrf-name*]	-
Configure the OSPF host route.	**host** *ip-address* **area** *area-id* [**cost** *cost*]	Mandatory. By default, no host route is generated.

6.2.6. Configure OSPF Route Control

Configuration Condition

Before configuring OSPF route control, ensure that:

- Interface IP addresses have been configured so that neighbor nodes are reachable at the network layer.
- Enable OSPF.

Configure Route Summary on Inter-Area OSPF Routes

When an ABR in OSPF advertises inter-area routes to other areas, it advertises each route separately in the form of Type-3 LSA. You can use the inter-area route summary function to summarize some continuous network segments to form a summary route. Then the ABR advertises the summary route, reducing the size of OSPF databases.

Table 6-15 Configure route summary on inter-area OSPF routes

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the OSPF configuration mode.	**router ospf** *process-id* [**vrf** *vrf-name*]	-
Configure route summary on inter-area OSPF routes.	**area** *area-id* **range** *ip-address/mask-length* [**advertise** [**cost** *cost*] | **cost** *cost* | **not-advertise**]	Mandatory. By default, an ABR does not summarize inter-area routes.

Note:

- The OSPF inter-area route summary function is valid only for ABRs.
- By default, the minimum cost value among the cost values of the routes in the route summary is used as the cost value of the route summary.

Configure OSPF External Route Summary

When OSPF redistributes external routes, it advertises each route separately in the form of external LSA. You can use the external route summary function to summarize some continuous network segments to form a summary route. Then OSPF advertises the summary route, reducing the size of OSPF databases.

If you run the **summary-address** command on an ASBR, you can summarize all Type-5 LSAs and Type-7 LSAs within the address range.

Table 6-16 Configure OSPF external route summary

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the OSPF configuration mode.	**router ospf** *process-id* [**vrf** *vrf-name*]	-

Step	Command	Description
Configure OSPF to summarize external routes.	**summary-address** *ip-address mask* [**not-advertise** | **tag** *tag-value*]	Mandatory. By default, an ABR does not summarize external routes.

Note:

- The OSPF external route summary function is valid only for ASBRs.

Configure Route Filtering on Inter-Area OSPF Routes

When an ABR receives inter-area routes, it performs filtration in the incoming direction based on an ACL or prefix list. When the ABR advertises inter-area routes, it performs filtration in the outgoing direction based on an ACL or prefix list.

Table 6-17 Configure route filtering on inter-area OSPF routes

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the OSPF configuration mode.	**router ospf** *process-id* [**vrf** *vrf-name*]	-
Configure route filtering on intra-area OSPF routes.	**area** *area-id* **filter-list** { **access** { *access-list-name* | *access-list-number* } | **prefix** *prefix-list-name* } { **in** | **out** }	Mandatory. By default, an ABR does not filter inter-area routes.

Note:

- In filtration based on ACL, only a standard ACL is supported.
- The OSPF inter-area route filtering function is valid only for ABRs.

Configure OSPF External Route Filtration

Configuring OSPF external route filtering is to apply an ACL or prefix list to allow or not allow external routes of an OSPF AS to flood into the OSPF AS.

Table 6-18 Configure OSPF external route filtration

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-

Step	**Command**	**Description**
Enter the OSPF configuration mode.	**router ospf** *process-id* [**vrf** *vrf-name*]	-
Configure a distribution list to filter external routes.	**distribute-list** { *access-list-name* | *access-list-number* | **prefix** *prefix-list-name* } **out** [*routing-protocol* [*process-id*]]	Mandatory. By default, an ASBR does not filter external routes.

Note:

- In filtration based on ACL, only a standard ACL is supported.
- The OSPF external route filtering function is valid only for ASBRs.

Configure OSPF Route Installation Filtration

After OSPF calculates routes through LSA, to prevent certain routes from being added into the route table, OSPF filters the calculated OSPF routing information.

Three filtration methods are available:

- Filtration based on the prefix. An ACL or prefix list is used to filter the destination addresses of routes.
- Filtration based on the next hop. A prefix list is used to filter the next hops of the routes. You can also use a prefix list to filter both the destination addresses and next hops of the routes.
- Filtration of routes based on the routing policy.

Table 6-19 Configure OSPF route installation filtration

Step	**Command**	**Description**
Enter the global configuration mode.	**configure terminal**	-
Enter the OSPF configuration mode.	**router ospf** *process-id* [**vrf** *vrf-name*]	-
Configure OSPF to prohibit installed routes.	**distribute-list** { *access-list-name* | *access-lsit-number* | **gateway** *prefix-list-name1* | **prefix** *prefix-list-name2* [**gateway** *prefix-list-name3*] | **route-map** *route-map-name* } **in** [*interface-name*]	Mandatory. By default, the installed routes are not filtered.

Note:

- Filtration based on prefix, gateway, and route-map is mutual exclusive with filtration based on ACL. For example, if you have configured filtration based on prefix, you cannot configure filtration based on ACL again.
- Filtration based on route-map and prefix is mutual exclusive with filtration based on gateway.
- Filtration based on prefix and filtration based on gateway overwrite each other.

Configure the Cost Value of an OSPF Interface

By default, the cost of an OSPF interface is calculated based on the following formula: Reference bandwidth/Interface bandwidth.

Table 6-20 Configure the cost value of an OSPF interface

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the interface configuration mode.	**interface** *interface-name*	-
Configure the cost value of an OSPF interface.	**ip ospf** [*ip-address*] **cost** *cost-value*	Optional. By default, the cost value is calculated through the formula Reference bandwidth/Interface bandwidth.

Configure the OSPF Reference Bandwidth

The reference bandwidth of an interface is used to calculate the cost value of the interface. The default value is 100Mbit/s. The formula for calculating the cost value of the OSPF interface is: Reference bandwidth/Interface bandwidth. If the calculation result is larger than 1, use the integer part. If the calculation result is smaller than 1, use the value 1. Therefore, in a network whose bandwidth is larger than 100Mbit/s, the optimal route fails to be selected. In this case, you can use the **auto-cost reference-bandwidth** command to configure a proper reference bandwidth to solve the problem.

Table 6-21 Configure the OSPF reference bandwidth

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-

Step	Command	Description
Enter the OSPF configuration mode.	**router ospf** *process-id* [**vrf** *vrf-name*]	-
Configuring the OSPF interface reference bandwidth.	**auto-cost reference-bandwidth** *reference-bandwidth*	Optional. By default, the reference bandwidth is 100Mbit/s.

Configure the OSPF Administrative Distance

An administrative distance is used to indicate the reliability of the routing protocol. If the routes to the same destination network are learnt by different routing protocols, the route with the smallest administrative distance is selected with priority.

Table 6-22 Configure the OSPF administrative distance

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the OSPF configuration mode.	**router ospf** *process-id* [**vrf** *vrf-name*]	-
Configure the OSPF administrative distance.	**distance** { *distance* [*ip-address wildcard-mask* [*access-list-name* | *access-list-number*]] | **ospf** { **external** *distance* | **inter-area** *distance* | **intra-area** *distance* } }	Optional. By default, the administrative distance of intra-area and inter-area OSPF routes is 110, and the administrative distance of external routes is 150.

Configure the Maximum Number of OSPF Load Balancing Routes

If multiple equivalent paths are available to reach the same destination, load balancing is achieved. This improves the utility rate of links and reduces the load of the links.

Table 6-23 Configure the maximum number of OSPF load balancing routes

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the OSPF configuration mode.	**router ospf** *process-id* [**vrf** *vrf-name*]	-
Configure the maximum number of OSPF load balancing routes.	**maximum-path** *max-number*	Optional. By default, the maximum number of OSPF load balancing routes is 4.

Configure OSPF to Be Compatible with RFC1583

If there exist multiple paths to an ASBR or external route forwarding address, RFC1583 and RFC2328 define different routing rules. If OSPF is configured to be compatible with RFC1583, the intra-area or inter-area paths in the backbone area is selected with priority. If OSPF is configured not to be compatible with RFC1583, the intra-area paths in non-backbone networks are selected with priority.

Table 6-24 Configure OSPF to be compatible with RFC1583

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the OSPF configuration mode.	**router ospf** *process-id* [**vrf** *vrf-name*]	-
Configure OSPF to be compatible with RFC1583.	**compatible rfc1583**	Mandatory. By default, OSPF is not compatible with RFC1583.

Note:

- In an OSPF AS, the routing rules of all the devices must be the same, that is, they must be all configured to be compatible with or not compatible with RFC1583 to prevent routing loops.

6.2.7. Configure OSPF Network Optimization

Configuration Condition

Before configuring OSPF network optimization, ensure that:

- Interface IP addresses have been configured so that neighbor nodes are reachable at the network layer.
- Enable OSPF.

Configure the Keep-alive Time of an OSPF Neighbor

OSPF Hello packets are used to set up neighbor relations and keep the relations alive. The default transmission interval of Hello packets is determined by the network type. For broadcast networks and P2P networks, the default transmission interval of Hello packets is 10s. For P2MP networks and NBMA networks, the default transmission interval of Hello packets is 30s.

Neighbor dead time is used to determine the validity of a neighbor. By default, the neighbor dead time is four times the Hello interval. If an OSPF device fails to receive Hello packets from a neighbor after the neighbor dead time times out, the OSPF device regards the neighbor as invalid, and then it deletes the neighbor in an active manner.

Table 6-25 Configure the keep-alive time of an OSPF neighbor

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the interface configuration mode.	**interface** *interface-name*	-
Configure an OSPF Hello interval.	**ip ospf** [*ip-address*] **hello-interval** *interval-value*	Optional. The default value is determined by the network type. For broadcast networks and P2P networks, the default value is 10s. For P2MP networks and NBMA networks, the default value is 30s.
Configure the OSPF neighbor dead time.	**ip ospf** [*ip-address*] **dead-interval** *interval-value*	Optional. By default, the time is four times the Hello interval.

Note:

- The Hello interval and neighbor dead time of OSPF neighbors must be the same; otherwise, they cannot set up neighbor relations.
- When you modify the Hello interval, if the current neighbor dead time is four times the Hello interval, the neighbor dead time is automatically modified to be still four times the new Hello interval. If the current neighbor dead time is not four times the Hello interval, the neighbor dead time keeps unchanged.
- If you modify the neighbor dead time, the Hello interval is not affected.

Configure an OSPF Passive Interface

The dynamic routing protocol adopts a passive interface to effectively decrease the network bandwidth consumed by the routing protocol. After an OSPF passive interface is configured, you can use the **network** command to advertise the routes of the directly connected network segment in which the interface is located, but the receiving and transmitting of OSPF packets are damped on the interface.

Table 6-26 Configure an OSPF passive interface

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the OSPF configuration mode.	**router ospf** *process-id* [**vrf** *vrf-name*]	-
Configure an OSPF passive interface.	**passive-interface** { *interface-name* [*ip-address*] | **default** }	Mandatory. By default, no OSPF passive interface is configured.

Configure an OSPF Demand Circuit

On P2P and P2MP links, to decrease the line cost, you can configure an OSPF demand circuit to suppress periodical transmitting of Hello packets and periodical update of LSA packets. This function is mainly applied on charged links such as ISDN, SVC, and X.25.

Table 6-27 Configure an OSPF demand circuit

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the interface configuration mode.	**interface** *interface-name*	-

Step	Command	Description
Configure an OSPF demand circuit.	**ip ospf** [*ip-address*] **demand-circuit**	Mandatory. By default, no OSPF demand circuit is enabled.

Configure the Priority of an OSPF Interface

Interface priorities are mainly used in election of Designated Router (DR), and Backup Designated Router (BDR) in broadcast networks and NBMA networks. The value range is 0-255. The larger the value is, the higher the priority is. The default value is 1.

The DR and BDR are selected from all devices in a network segment based on interface priorities and Router IDs through Hello packets. The rules are as follows:

- First, the device whose interface has the highest priority is elected as the DR, and the device whose interface has the second highest priority is elected as the BDR. The device whose interface has the priority 0 does not participate in the election.
- If the interface priorities of two devices are the same, the device with the largest Router ID is elected as the DR, and the device with the second largest Router ID is elected as the BDR.
- If the DR fails, the BDR becomes the DR immediately, and a new BDR is elected.

Table 6-28 Configure the priority of an OSPF interface

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the interface configuration mode.	**interface** *interface-name*	-
Configure the priority of an OSPF interface.	**ip ospf priority** *priority-value*	Optional. By default, the OSPF interface priority is 1.

Note:

- Interface priorities affect only an election process. If the DR and BDR have already been elected, modification of interface priorities does not affect the election result; instead, it affects the next election of DR or BDR. Therefore, the DR may not have the interface with the highest priority, and the BDR may not have the interface with the second highest priority.

Configure the MTU of an OSPF Interface

In encapsulating OSPF packets, to prevent fragmentation, you need to limit the packet size to equal to or smaller than Maximum Transmission Unit (MTU) of the interface. When adjacent

OSPF devices exchange DD packets, MTUs are checked by default. If the MTUs are different, the devices cannot form a neighbor relation. If you have configured OSPF to ignore interface MTU check, even if MTUs are different, they can set up a neighbor relation.

Table 6-29 Configure the MTU of an OSPF interface

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the interface configuration mode.	**interface** *interface-name*	-
Configure the MTU of an OSPF interface.	**ip ospf mtu** *mtu-value*	Optional.
Configure the OSPF interface to ignore MTU consistency check.	**ip ospf** [*ip-address*] **mtu-ignore**	Mandatory. By default, an MTU consistency check will be performed.

Configure the LSA Transmit Delay of an OSPF Interface

LSA transmit delay refers to the time it takes for an LSA to flood to other devices. The device that sends the LSA adds the interface transmit delay to the LSA aging time. By default, once the flooding LSA passes a device, the aging time is increased by 1. You can configure the LSA transmit delay according to the network conditions. The value range is 1-840. LSA transmit delay is usually configured on low-speed links.

Table 6-30 Configure the LSA transmit delay of an OSPF interface

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the interface configuration mode.	**interface** *interface-name*	-
Configure the LSA transmit delay of an OSPF interface.	**ip ospf transmit-delay** *delay-value*	Optional. By default, the LSA transmit delay is 1s.

Configure OSPF LSA Retransmission

To ensure the reliability of data exchange, OSPF adopts the acknowledgement mechanism. If an LSA floods on a device interface, the LSA is added into the retransmission list of the neighbor. If no acknowledgement message is received from the neighbor after the retransmission time times out, the LSA is retransmitted until an acknowledgement message is received.

After the re-transmission queue scan timer times out, the LSA meeting the re-transmission time in the re-transmission queue is filled to the LSU packet and sent out, sending one LSU packet every time. The LSA re-transmission time can be modified by the **ip ospf retransmit-interval** command. By default, the initial re-transmission time is 5s. You can control the time parameter of the re-transmission queue scan timer by miodifying the scan interval of the LSA re-transmission queue. By default, the scan interval of the LSA re-transmission queue is 1250ms.

Table 6-31 Configure OSPF LSA retransmission

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the interface configuration mode.	**interface** *interface-name*	-
Configure the interval of OSPF LSA retransmission.	**ip ospf retransmit-interval** *interval-value*	Optional. By default, the retransmission interval is 5s.
Enter the global configuration mode	**exit**	-
Enter the OSPF configuration mode	**router ospf** *process-id* [**vrf** *vrf-name*]	-
Configure the scan interval of the OSPF LSA re-transmission queue	**timers lsa retransmission** *scan-time*	Optional By default, the scan interval of the re-transmission queue is 1250ms.

Configure OSPF LSA Sending

When the device interface prepares to flood one LSA, add the LSA to the sending queue of the interface. After the scan timer of the interface sending queue times out, the LSA in the queue is filled to the LSU packet and sent out, sending one LSU packet every time. You can control the time parameter of the sending queue scan timer by miodifying the LSA sending queue. By default, the scan interval of the LSA sending queue is 100ms.

Table 6-32 Configure the OSPF LSA sending

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the interface configuration mode.	**interface** *interface-name*	-
Configure the scan interval of the OSPF specified interface sending queue	**ip ospf lsa-interval** *interval-value* [**auto**]	Optional By default, the scan interval of the interface sending queue is 100ms.
Enter the global configuration mode	**exit**	-
Enter the OSPF configuration mode	**router ospf** *process-id* [**vrf** *vrf-name*]	-
Configure the scan interval of all interface sending queues of the OSPF process	**timers lsa transmission** *scan-time* [**auto**]	Optional By default, the scan interval of the sending queue is 100ms.

Note:

- When configuring the scan interval of the sending queue in the OSPF configuration mode and the interface configuration mode at the same time, the configuration in the interface has higher priority.

Configure OSPF to Prevent LSA Flooding

In actual network applications, redundant links may be used between OSPF neighbors under some circumstances. This configuration helps to decrease flooding of OSPF update packets on redundant links.

Table 6-33 Configure OSPF to prevent LSA flooding

Step	**Command**	**Description**
Enter the global configuration mode.	**configure terminal**	-
Enter the interface configuration mode.	**interface** *interface-name*	-
Configure the OSPF interface to prevent LS-UPD flooding.	**ip ospf database-filter all out**	Mandatory. By default, the OSPF interface does not prevent LSA flooding.

Note:

- Configuring OSPF to prevent LSA spreading may result in loss of some routing information.

Configure OSPF SPF Calculation Time

If the OSPF network topology changes, routes need to be re-calculated. When the network continues to change, frequent route calculation occupies a lot of system resources. You can adjust the SPF calculation time parameters to prevent frequent network changes from consuming too many system resources.

Table 6-34 Configure OSPF SPF calculation time

Step	**Command**	**Description**
Enter the global configuration mode.	**configure terminal**	-
Enter the OSPF configuration mode.	**router ospf** *process-id* [**vrf** *vrf-name*]	-
Configure OSPF SPF calculation time.	**timers throttle spf** *delay-time hold-time max-time*	Optional. By default, *delay-time* is 5000ms, *hold-time* is 10000ms, and *max-time* is 10000ms.

Note:

- The parameter *delay-time* indicates the initial calculation delay, *hold-time* indicates the suppression time, and *max-time* indicates the maximum waiting time between two SPF calculations. If network changes are not frequent, you can shorten the continuous route calculation interval to *delay-time*. If network changes are frequent, you can adjust the parameters, increase the suppression time to *hold-time*×2^{n-2} (n is the number of route calculation trigger times), extend the waiting time based on the configured *hold-time* increment and the maximum value must not exceed *max-time*.

Configure OSPF Database Overflow

OSPF database overflow is used to limit the number of Type-5 LSAs in the database.

Table 6-35 Configure OSPF Databases overflow

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the OSPF configuration mode.	**router ospf** *process-id* [**vrf** *vrf-name*]	-
Configure OSPF database overflow.	**overflow database external** *max-number seconds*	Mandatory. By default, the OSPF database overflow function is disabled.

Caution:

- After the database overflow function is enabled, the databases in the OSPF area may become inconsistent, and some routes get lost.

6.2.8. Configure OSPF to Coordinate with BFD

Configuration Condition

Before configuring OSPF to coordinate with BFD, ensure that:

- Interface IP addresses have been configured so that neighbor nodes are reachable at the network layer.
- Enable OSPF.

Configure OSPF to Coordinate with BFD

Bidirectional Forwarding Detection (BFD) provides a method for quickly detecting the status of a line between two devices. If BFD is started between two adjacent OSPF devices, if the line between two devices becomes faulty, BFD quickly detects the fault and informs OSPF of the fault. Then, it triggers OSPF to start route calculation and switch over to the backup line, achieving fast switchover of routes.

Table 6-36 Configure OSPF to coordinate with BFD

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the interface configuration mode.	**interface** *interface-name*	-
Enable or disable BFD on the specified OSPF interface.	**ip ospf bfd** [*ip-address* | **disable**]	Mandatory. By default, the BFD function is disabled.
Enter the global configuration mode.	**exit**	-
Enter the OSPF configuration mode.	**router ospf** *process-id* [**vrf** *vrf-name*]	-
Enable BFD on all interfaces of the OSPF process.	**bfd all-interface**	Optional.

Note:

- If BFD is configured both in OSPF configuration mode and interface configuration mode, the BFD configuration on the interface has the higher priority.

6.2.9. Configure OSPF Fast Re-routing

Configuration Condition

Before configuring OSPF fast re-routing, ensure that:

- Interface IP addresses have been configured so that neighbor nodes are reachable at the network layer.
- Enable OSPF.

Configure OSPF Fast Re-routing

In the OSPF network, if the link or device fails, the packet passing the fault point will be dropped or generate the loop and the caused traffic interruption will not recover until the protocol re-converges, which often lasts for several seconds. To reduce the traffic interruption time, you can configure the OSPF fast re-routing. Apply the route map to set the backup next hop for the matched route. Once the active link fails, the traffic passing the faulty link will switch to the standby link at once.

Table 6-37 Configure the OSPF fast re-routing

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the OSPF configuration mode.	**router ospf** *process-id* [**vrf** *vrf-name*]	-
Configure the OSPF process to enable the fast re-routing function	**fast-reroute route-map** *route-map-name*	Mandatory By default, do not enable the OSPF fast re-routing function.
Configure the OSPF process to enable the dynamic fast re-routing function	**fast-reroute loop-free-alternate** [**route-map** *route-map-name*]	Mandatory By default, do not enable the OSPF dynamic fast re-routing function.
Configure the OSPF process to enable the pic function	**pic**	Mandatory After enabling the pic function, enable the auto fast re-routing function. By default, do not enable the OSPF pic function.

6.2.10. OSPF Monitoring and Maintaining

Table 6-38 OSPF monitoring and maintaining

Command	Description
clear ip ospf [*process-id*] **process**	Reset an OSPF process.
clear ip ospf *process-id* **neighbor** *neighbor-ip-address* [*neighbor-router-id*]	Reset an OSPF neighbor.

Command	Description
clear ip ospf err-statistic	Clear the OSPF error statistics information.
clear ip ospf statistics [*interface-name*]	Clear OSPF interface statistics.
clear ip ospf [*process-id*] **redistribution**	Re-advertises external routes.
clear ip ospf [*process-id*] **route**	Re-calculate OSPF routes.
show ip ospf [*process-id*]	Display the OSPF basic information.
show ip ospf [*process-id*] **border-routers**	Display the information about the routes that have reached the boundary devices in OSPF.
show ip ospf [*process-id*] **buffers**	Display the size of the OSPF packet transmitting and receiving buffer.
show ip ospf [*process-id*] **calc-inrement-list**	Display the LSA increment calculation list
show ip ospf [*process-id*] **database** [**adv-router** *router-id* | **age** *lsa_age* | **database-summary | max-age |** [**asbr-summary | external | network | nssa-external | opaque-area | opaque-as | opaque-link | router | self-originate | summary**] [[*link-state-id*] [**adv-router** *advertising-router-id*] | **self-originate | summary**]]	Display the information about an OSPF database.
show ip ospf error-statistic	Display the OSPF error statistics information
show ip ospf interface [*interface-name* [**detail**]]	Display the information about an OSPF interface.

Command	Description
show ip ospf [*process-id*] **neighbor** [*neighbor-id* | **all** | **detail** [**all**] | **interface** *ip-address* [**detail**] | **statistic**]	Display the information about OSPF neighbors.
show ip ospf *process-id* **retransmit-list**	Display the retransmission list information
show ip ospf route [*ip-address mask* | *ip-address/mask-length* | **external** | **inter-area** | **intra-area** | **statistic**]	Display the information about OSPF routes.
show ip ospf [*process-id*] **virtual-links**	Display the information about OSPF virtual links.
show ip ospf [*process-id*] **sham-links**	Display the information about an interface on which OSPF sham links are configured. The information include interface status, cost value, and neighbor status.

6.3. OSPF Typical Configuration Example

6.3.1. Configure Basic OSPF Functions

Network Requirements

- Configure OSPF for all devices, and divide the devices into three areas: Area 0, Area 1, and Area 2. After configuration, all devices should be able to learn routes from each other.
- On a back-to-back Ethernet interface, to speed up set of OSPF neighbors, you can change the network type of the OSPF interface to P2P. Modify the network type of the interfaces in Area 2 to P2P. After the configuration, all devices can learn routes from each other.

Network Topology

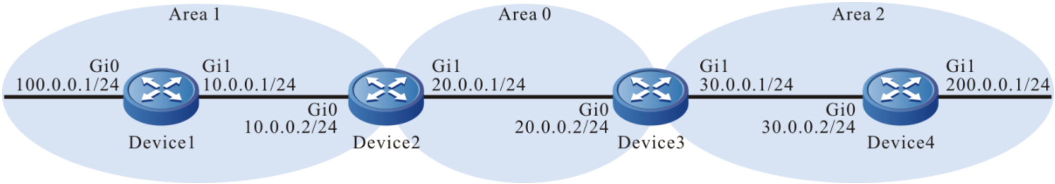

Figure 6-1 Networking for configuring basic OSPF functions

Configuration Steps

Step 1: Configure the IP addresses of the interfaces. (Omitted)

Step 2: Configure an OSPF process and let the interface cover different areas.

#On Device1, configure an OSPF process and configure the interfaces to cover area 1.

```
Device1#configure terminal
Device1(config)#router ospf 100
Device1(config-ospf)#router-id 1.1.1.1
Device1(config-ospf)#network 10.0.0.0 0.0.0.255 area 1
Device1(config-ospf)#network 100.0.0.0 0.0.0.255 area 1
Device1(config-ospf)#exit
```

#On Device2, configure an OSPF process and configure the interfaces to cover Area 0 and Area 1.

```
Device2#configure terminal
Device2(config)#router ospf 100
Device2(config-ospf)#router-id 2.2.2.2
Device2(config-ospf)#network 20.0.0.0 0.0.0.255 area 0
Device2(config-ospf)#network 10.0.0.0 0.0.0.255 area 1
Device2(config-ospf)#exit
```

#On Device3, configure an OSPF process and configure the interfaces to cover Area 0 and Area 2.

```
Device3#configure terminal
Device3(config)#router ospf 100
Device3(config-ospf)#router-id 3.3.3.3
Device3(config-ospf)#network 20.0.0.0 0.0.0.255 area 0
Device3(config-ospf)#network 30.0.0.0 0.0.0.255 area 2
Device3(config-ospf)#exit
```

#On Device4, configure an OSPF process and configure the interfaces to cover area 2.

```
Device4#configure terminal
Device4(config)#router ospf 100
Device4(config-ospf)#router-id 4.4.4.4
Device4(config-ospf)#network 30.0.0.0 0.0.0.255 area 2
Device4(config-ospf)#network 200.0.0.0 0.0.0.255 area 2
Device4(config-ospf)#exit
```

Note:

- A Router IDs can be manually configured or automatically generated. If a Router ID is not manually configured, the device automatically selects a Router ID. The device first selects the largest IP address among Loopback interface IP addresses as the Router ID. If the device is not configured with Loopback interface IP addresses, then it selects the largest IP addresses among common interface IP addresses as the Router ID. Only when an interface is in the UP status can the IP address of the interface be selected as the Router ID.
- In using the **network** command, the wildcard mask need not accurately match the mask length of the interface IP addresses, but the network segment needs to cover the

interface IP addresses. For example, network 0.0.0.0 255.255.255.255 means to cover all interfaces.

#Query the OSPF neighbors and route table of Device1.

```
Device1#show ip ospf neighbor
OSPF process 100:
Neighbor ID     Pri   State           Dead Time   Address         Interface
2.2.2.2           1   Full/DR         00:00:36    10.0.0.2        gigabitethernet1

Device1#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

C   10.0.0.0/24 is directly connected, 02:26:21, gigabitethernet1
L   10.0.0.1/32 is directly connected, 02:26:21, gigabitethernet1
O   20.0.0.0/24 [110/2] via 10.0.0.2, 02:25:36, gigabitethernet1
O   30.0.0.0/24 [110/3] via 10.0.0.2, 02:25:36, gigabitethernet1
C   100.0.0.0/24 is directly connected, 02:26:23, gigabitethernet0
L   100.0.0.1/32 is directly connected, 02:26:23, gigabitethernet0
C   127.0.0.0/8 is directly connected, 18:09:44, lo0
L   127.0.0.1/32 is directly connected, 18:09:44, lo0
O   200.0.0.0/24 [110/4] via 10.0.0.2, 02:25:36, gigabitethernet1
```

#Query the OSPF neighbors and route table of Device2.

```
Device2#show ip ospf neighbor
OSPF process 100:
Neighbor ID     Pri   State           Dead Time   Address         Interface
1.1.1.1         1   Full/Backup     00:00:37    10.0.0.1        gigabitethernet0
3.3.3.3           1   Full/DR         00:00:38    20.0.0.2        gigabitethernet1

Device2#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

C   10.0.0.0/24 is directly connected, 02:31:15, gigabitethernet0
L   10.0.0.2/32 is directly connected, 02:31:15, gigabitethernet0
C   20.0.0.0/24 is directly connected, 02:31:50, gigabitethernet1
```

```
L   20.0.0.1/32 is directly connected, 02:31:50, gigabitethernet1
O   30.0.0.0/24 [110/2] via 20.0.0.2, 02:31:40, gigabitethernet1
O   100.0.0.0/24 [110/2] via 10.0.0.1, 02:30:29, gigabitethernet0
C   127.0.0.0/8 is directly connected, 240:21:34, lo0
L   127.0.0.1/32 is directly connected, 240:21:34, lo0
O   200.0.0.0/24 [110/3] via 20.0.0.2, 02:31:40, gigabitethernet1
```

#Query OSPF Link Status Database (LSDB) of Device2.

```
Device2#show ip ospf database

            OSPF Router with ID (2.2.2.2) (Process ID 100)

                Router Link States (Area 0)

Link ID         ADV Router      Age Seq#       CkSum  Link count
2.2.2.2         2.2.2.2         1777 0x8000000c 0xcb20 1
3.3.3.3         3.3.3.3          309 0x8000000a 0x9153 1

                Net Link States (Area 0)

Link ID         ADV Router      Age Seq#       CkSum
20.0.0.2        3.3.3.3          369 0x80000006 0xec12

                Summary Link States (Area 0)

Link ID         ADV Router      Age Seq#       CkSum  Route
10.0.0.0        2.2.2.2         1757 0x80000005 0xcc59 10.0.0.0/24
100.0.0.0       2.2.2.2         1356 0x80000005 0x408a 100.0.0.0/24
30.0.0.0        3.3.3.3            9 0x80000006 0xa765 30.0.0.0/24
200.0.0.0       3.3.3.3          149 0x80000006 0x075a 200.0.0.0/24

                Router Link States (Area 1)

Link ID         ADV Router      Age Seq#       CkSum  Link count
1.1.1.1         1.1.1.1         1775 0x80000009 0xbbda 2
2.2.2.2         2.2.2.2         1737 0x80000008 0x2dd5 1

                Net Link States (Area 1)
```

```
Link ID       ADV Router      Age Seq#       CkSum
10.0.0.2      2.2.2.2          34 0x80000006 0x39db

                Summary Link States (Area 1)

Link ID       ADV Router      Age Seq#       CkSum  Route
20.0.0.0      2.2.2.2         144 0x80000006 0x48d2 20.0.0.0/24
30.0.0.0      2.2.2.2        1186 0x80000005 0xd13f 30.0.0.0/24
200.0.0.0     2.2.2.2          14 0x80000006 0x2f35 200.0.0.0/24
```

For Device2, routes 30.0.0.0/24 and 200.0.0.0/24 are inter-area routes. You can query the LSA information of the related routes in Summary Link States (Area 0). In the case of intra-area routes, run the **show ip ospf database router** command to query the LSA information of the related routes.

Step 3: Configure the network type of OSPF interfaces to P2P.

#On Device3, configure the OSPF network type of interface gigabitethernet 0/2/1 to P2P.

```
Device3(config)#interface gigabitethernet1
Device3(config-if-gigabitethernet1)#ip ospf network point-to-point
Device3(config-if-gigabitethernet1)#exit
```

#On Device4, configure the OSPF network type of interface gigabitethernet 0 to P2P.

```
Device4(config)#interface gigabitethernet0
Device4(config-if-gigabitethernet0)#ip ospf network point-to-point
Device4(config-if-gigabitethernet0)#exit
```

Step 4: Check the result.

#Query the OSPF neighbors and route table of Device3.

```
Device3#show ip ospf neighbor
OSPF process 100:
Neighbor ID     Pri   State           Dead Time   Address         Interface
2.2.2.2           1   Full/Backup     00:00:36    20.0.0.1        gigabitethernet0
4.4.4.4           1   Full/ -         00:00:39    30.0.0.2        gigabitethernet1

Device3#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

O   10.0.0.0/24 [110/2] via 20.0.0.1, 00:02:53, gigabitethernet0
```

```
C   20.0.0.0/24 is directly connected, 03:20:36, gigabitethernet0
L   20.0.0.2/32 is directly connected, 03:20:36, gigabitethernet0
C   30.0.0.0/24 is directly connected, 03:20:26, gigabitethernet1
L   30.0.0.1/32 is directly connected, 03:20:26, gigabitethernet1
O   100.0.0.0/24 [110/3] via 20.0.0.1, 00:01:51, gigabitethernet0
C   127.0.0.0/8 is directly connected, 262:01:24, lo0
L   127.0.0.1/32 is directly connected, 262:01:24, lo0
O   200.0.0.0/24 [110/2] via 30.0.0.2, 00:00:11, gigabitethernet1
```

Note:

- If OSPF neighbor relations are set up in a P2P network, no DR or BDR election will be performed.

#Query the OSPF neighbors and route table of Device4.

```
Device4#show ip ospf neighbor
OSPF process 100:
Neighbor ID     Pri   State          Dead Time   Address        Interface
3.3.3.3           1   Full/ -        00:00:39    30.0.0.1       gigabitethernet0

Device4#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

O   10.0.0.0/24 [110/3] via 30.0.0.1, 00:01:04, gigabitethernet0
O   20.0.0.0/24 [110/2] via 30.0.0.1, 00:01:04, gigabitethernet0
C   30.0.0.0/24 is directly connected, 03:20:25, gigabitethernet0
L   30.0.0.2/32 is directly connected, 03:20:25, gigabitethernet0
O   100.0.0.0/24 [110/4] via 30.0.0.1, 00:01:04, gigabitethernet0
C   127.0.0.0/8 is directly connected, 22:52:36, lo0
L   127.0.0.1/32 is directly connected, 22:52:36, lo0
C   200.0.0.0/24 is directly connected, 03:20:13, gigabitethernet1
L   200.0.0.1/32 is directly connected, 03:20:13, gigabitethernet1
```

After the network type of OSPF interfaces are modified to P2P, neighbors can be set up normally, and routes can be learned normally.

Note:

- In configuring network types for OSPF interfaces, the network types of OSPF interfaces at the two ends of neighbors must be the same; otherwise, routing learning and flooding will be affected. By default, the network type of an OSPF interface is determined by the network type of the physical interface.

6.3.2. Configuring OSPF Authentication

Network Requirements

- Configure OSPF for all devices run OSPF, and configure area authentication for the devices. Configure simple text authentication for Area 0, and configure MD5 authentication for Area 1.
- Configure OSPF interface authentication, configure interface authentication of Area 0 to simple text authentication, and configure interface authentication of Area 1 to MD5 authentication.
- After configuration is completed, devices should able to normally set up neighbor relations and learn route from each other.

Network Topology

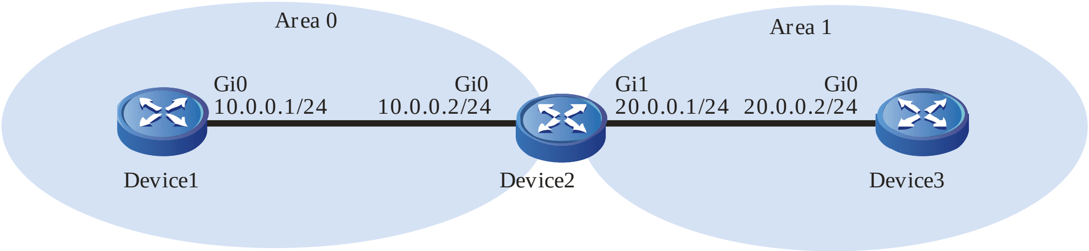

Figure 6-2 Networking for configuring OSPF authentication

Configuration Steps

Step 1: Configure the IP addresses of the interfaces. (Omitted)

Step 2: Configure an OSPF process, and configure the interfaces to cover different areas, and enable area authentication. Configure the simple text authentication for Area 0, and configure the MD5 authentication for Area 1.

#On Device1, configure an OSPF process and configure the area authentication function.

```
Device1#configure terminal
Device1(config)#router ospf 100
Device1(config-ospf)#router-id 1.1.1.1
Device1(config-ospf)#area 0 authentication
Device1(config-ospf)#network 10.0.0.0 0.0.0.255 area 0
Device1(config-ospf)#exit
```

#On Device2, configure an OSPF process and configure the area authentication function.

```
Device2#configure terminal
Device2(config)#router ospf 100
Device2(config-ospf)#router-id 2.2.2.2
Device2(config-ospf)#area 0 authentication
Device2(config-ospf)#area 1 authentication message-digest
Device2(config-ospf)#network 10.0.0.0 0.0.0.255 area 0
Device2(config-ospf)#network 20.0.0.0 0.0.0.255 area 1
```

```
Device2(config-ospf)#exit
```

#On Device3, configure an OSPF process and configure the area authentication function.

```
Device3#configure terminal
Device3(config)#router ospf 100
Device3(config-ospf)#router-id 3.3.3.3
Device3(config-ospf)#area 1 authentication message-digest
Device3(config-ospf)#network 20.0.0.0 0.0.0.255 area 1
Device3(config-ospf)#exit
```

#Query the OSPF process information of Device1.

```
Device1#show ip ospf 100
 Routing Process "ospf 100" with ID 1.1.1.1
 Process bound to VRF default
 Process uptime is 18 minutes
 IETF NSF restarter support disabled
 IETF NSF helper support enabled
 Conforms to RFC2328, and RFC1583Compatibility flag is disabled
 Supports only single TOS(TOS0) routes
 Supports opaque LSA
 Supports Graceful Restart
 Initial SPF schedule delay 5000 msecs
 Minimum hold time between two consecutive SPFs 10000 msecs
 Maximum wait time between two consecutive SPFs 10000 msecs
ASE route calculate executed 0 timesRefresh timer 10 secs
 Transmit timer 100 msecs, burst num 1 Number of external LSA 0. Checksum Sum
0x000000
 Number of opaque AS LSA 0. Checksum Sum 0x000000
 Number of non-default external LSA is 0
 External LSA database is unlimited.
 Not Support Demand Circuit lsa number is 0, autonomy system support flood
DoNotAge Lsa
 Number of areas attached to this router: 1
    Area 0 (BACKBONE)       Number of interfaces in this area is 1(1)
        Number of fully adjacent neighbors in this area is 0
        Number of fully adjacent sham-link neighbors in this area is 0
        Area has simple password authentication
        SPF algorithm last executed 00:16:35.783 ago
        SPF algorithm executed 3 times
        IA route calculat executed 0 times        Number of LSA 2. Checksum Sum
0x010154
        Not Support Demand Circuit lsa number is 0,
        Indication lsa (by other routers) number is: 0,
        Area support flood DoNotAge Lsa
```

According to the queried information, the area authentication is the simple text mode.

#Query the OSPF neighbors and route table of Device1.

```
Device1#show ip ospf neighbor
OSPF process 100:
Neighbor ID     Pri   State          Dead Time   Address        Interface
2.2.2.2           1   Full/DR     00:00:38    10.0.0.2       gigabitethernet0

Device1#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

C   10.0.0.0/24 is directly connected, 00:14:01, gigabitethernet0
L   10.0.0.1/32 is directly connected, 00:14:01, gigabitethernet0
O   20.0.0.0/24 [110/2] via 10.0.0.2, 00:10:38, gigabitethernet0
C   127.0.0.0/8 is directly connected, 20:55:08, lo0
L   127.0.0.1/32 is directly connected, 20:55:08, lo0
```

On Device1, neighbors can be normally set up, and routes can be learnt normally.

#Query the OSPF process information of Device3.

```
Device3#show ip ospf 100
 Routing Process "ospf 100" with ID 3.3.3.3
 Process bound to VRF default
 Process uptime is 2 minutes
 IETF NSF restarter support disabled
 IETF NSF helper support enabled
 Conforms to RFC2328, and RFC1583Compatibility flag is disabled
 Supports only single TOS(TOS0) routes
 Supports opaque LSA
 Supports Graceful Restart
 Initial SPF schedule delay 5000 msecs
 Minimum hold time between two consecutive SPFs 10000 msecs
 Maximum wait time between two consecutive SPFs 10000 msecs
 ASE route calculate executed 0 times Refresh timer 10 secs
 Transmit timer 100 msecs, burst num 1 Number of external LSA 0. Checksum Sum
0x000000
 Number of opaque AS LSA 0. Checksum Sum 0x000000
 Number of non-default external LSA is 0
 External LSA database is unlimited.
 Not Support Demand Circuit lsa number is 0, autonomy system support flood
DoNotAge Lsa
 Number of areas attached to this router: 1
    Area 1        Number of interfaces in this area is 1(1)
        Number of fully adjacent neighbors in this area is 1
        Number of fully adjacent sham-link neighbors in this area is 0
        Number of fully adjacent virtual neighbors through this area is 0
        Area has message digest authentication
        SPF algorithm last executed 00:00:22.500 ago
        SPF algorithm executed 3 times
        IA route calculat last executed 00:00:22.500 ago
        IA route calculat executed 1 times
        Number of LSA 4. Checksum Sum 0x01cf59
        Not Support Demand Circuit lsa number is 0,
        Indication lsa (by other routers) number is: 0,
        Area support flood DoNotAge Lsa
```

According to the queried information, the area authentication is the MD5 authentication mode.

#Query the OSPF neighbors and route table of Device3.

```
Device3#show ip ospf neighbor
OSPF process 100:
Neighbor ID     Pri   State          Dead Time   Address        Interface
2.2.2.2          1   Full/Backup     00:00:33    20.0.0.1       gigabitethernet0

Device3#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

O   10.0.0.0/24 [110/2] via 20.0.0.1, 00:09:31, gigabitethernet0
C   20.0.0.0/24 is directly connected, 00:20:36, gigabitethernet0
L   20.0.0.2/32 is directly connected, 00:20:36, gigabitethernet0
C   127.0.0.0/8 is directly connected, 24:00:06, lo0
```

```
L   127.0.0.1/32 is directly connected, 24:00:06, lo0
```

On Device3, neighbors can be normally set up, and routes can be learnt normally.

Step 3: Configure OSPF interface authentication.

#On Device1, configure interface gigabitethernet 0 with simple text authentication, and set the password to admin.

```
Device1(config)#interface gigabitethernet0
Device1(config-if-gigabitethernet0)#ip ospf authentication
Device1(config-if-gigabitethernet0)#ip ospf authentication-key 0 admin
Device1(config-if-gigabitethernet0)#exit
```

#On Device2, configure interface gigabitethernet 0 with simple text authentication, and set the password to admin. Configure interface gigabitethernet 1 with MD5 authentication, set Key ID to 1, and set password to admin.

```
Device2(config)#interface gigabitethernet0
Device2(config-if-gigabitethernet0)#ip ospf authentication
Device2(config-if-gigabitethernet0)#ip ospf authentication-key 0 admin
Device2(config-if-gigabitethernet0)#exit
Device2(config)#interface gigabitethernet1
Device2(config-if-gigabitethernet1)#ip ospf authentication message-digest
Device2(config-if-gigabitethernet1)#ip ospf message-digest-key 1 md5 0 admin
Device2(config-if-gigabitethernet1)#exit
```

#On Device3, configure interface gigabitethernet 0 with MD5 authentication, set Key ID to 1, and set password to admin.

```
Device3(config)#interface gigabitethernet0
Device3(config-if-gigabitethernet0)#ip ospf authentication message-digest
Device3(config-if-gigabitethernet0)#ip ospf message-digest-key 1 md5 0 admin
Device3(config-if-gigabitethernet0)#exit
```

Step 4: Check the result.

#Query the OSPF neighbor information of Device2.

```
Device2#show ip ospf neighbor
OSPF process 100:
Neighbor ID     Pri   State          Dead Time   Address        Interface
1.1.1.1          1   Full/Backup    00:00:33    10.0.0.1       gigabitethernet0
3.3.3.3          1   Full/DR        00:00:39    20.0.0.2       gigabitethernet1
```

#Query the OSPF interface information of Device2.

```
Device2#show ip ospf interface gigabitethernet0
Gigabitethernet0 is up, line protocol is up
 Internet Address 10.0.0.2, 10.0.0.255( a[10.0.0.2] d[10.0.0.255]) Area 0, MTU 1500
```

```
  Process ID 100, Router ID 2.2.2.2, Network Type BROADCAST, Cost: 1
  Transmit Delay is 1 sec, State DR, Priority 1, TE Metric 0
  Designated Router (ID) 2.2.2.2, Interface Address 10.0.0.2
  Backup Designated Router (ID) 1.1.1.1, Interface Address 10.0.0.1
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5 , Transmit 100
msecs
    Hello due in 00:00:04
Transmisson List 0  Neighbor Count is 1, Adjacent neighbor count is 1
  Crypt Sequence Number is 0

  Graceful restart proxy id is 0x0  Hello received 406 sent 454, DD received 8 sent 6
  LS-Req received 2 sent 2, LS-Upd received 11(LSA: 15) sent 10(LSA: 14)
  LS-Ack received 10 sent 0, Discarded 0

Device2#show ip ospf interface gigabitethernet1
Gigabitethernet1 is up, line protocol is up
  Internet Address 20.0.0.1, 20.0.0.255( a[20.0.0.1] d[20.0.0.255]) Area 1, MTU 1500
  Process ID 100, Router ID 2.2.2.2, Network Type BROADCAST, Cost: 1
  Transmit Delay is 1 sec, State Backup, Priority 1, TE Metric 0
  Designated Router (ID) 3.3.3.3, Interface Address 20.0.0.2
  Backup Designated Router (ID) 2.2.2.2, Interface Address 20.0.0.1
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5 , Transmit 100
msecs
    Hello due in 00:00:00
Transmisson List 0  Neighbor Count is 1, Adjacent neighbor count is 1
  Crypt Sequence Number is 485

  Graceful restart proxy id is 0x0  Hello received 412 sent 454, DD received 9 sent 12
  LS-Req received 3 sent 3, LS-Upd received 9(LSA: 10) sent 13(LSA: 16)
  LS-Ack received 13 sent 8, Discarded 0
```

After MD5 authentication is configured, a Crypt Sequence Number is generated. In the case of simple text authentication, no sequence number is generated.

Note:

- In configuring OSPF authentication, you can configure only area authentication or interface authentication, or configure both of them.
- If both area authentication and interface authentication are configured, interface authentication takes effect first.

6.3.3. Configuring OSPF to Redistribute Routes

Network Requirements

- Run OSPF between Device1 and Device2, and run RIPv2 between Device2 and Device3.
- Device2 redistributes RIP routes to OSPF, and it uses a route policy to control the device to only redistribute route 100.0.0.0/24.

Network Topology

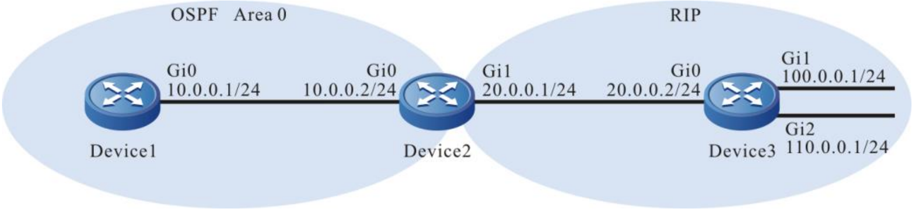

Figure 6-3 Networking for configuring OSPF to redistribute routes

Configuration Steps

Step 1: Configure the IP addresses of the interfaces. (Omitted)

Step 2: Configure OSPF between Device and Device2. Configure RIPv2 between Device2 and Device3.

#Configure OSPF for Device1.

```
Device1#configure terminal
Device1(config)#router ospf 100
Device1(config-ospf)#router-id 1.1.1.1
Device1(config-ospf)#network 10.0.0.0 0.0.0.255 area 0
Device1(config-ospf)#exit
```

#Configure OSPF and RIPv2 for Device2.

```
Device2#configure terminal
Device2(config)#router ospf 100
Device2(config-ospf)#router-id 2.2.2.2
Device2(config-ospf)#network 10.0.0.0 0.0.0.255 area 0
Device2(config-ospf)#exit
Device2(config)#router rip
Device2(config-rip)#version 2
Device2(config-rip)#network 20.0.0.0
Device2(config-rip)#exit
```

#Configure RIPv2 for Device3.

```
Device3#configure terminal
Device3(config)#router rip
Device3(config-rip)#version 2
Device3(config-rip)#network 20.0.0.0
Device3(config-rip)#network 100.0.0.0
```

```
Device3(config-rip)#network 110.0.0.0
Device3(config-rip)#exit
```

#Query the OSPF neighbor information of Device1.

```
Device1#show ip ospf neighbor
OSPF process 100:
Neighbor ID     Pri   State          Dead Time   Address        Interface
2.2.2.2           1   Full/DR        00:00:32    10.0.0.2       gigabitethernet0
```

#Query the OSPF neighbor information of Device2.

```
Device2#show ip ospf neighbor
OSPF process 100:
Neighbor ID     Pri   State          Dead Time   Address        Interface
1.1.1.1           1   Full/Backup    00:00:32    10.0.0.1       gigabitethernet0
```

#Query the route table of Device2.

```
Device2#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

C   10.0.0.0/24 is directly connected, 00:21:17, gigabitethernet0
L   10.0.0.2/32 is directly connected, 00:21:17, gigabitethernet0
C   20.0.0.0/24 is directly connected, 00:21:33, gigabitethernet1
L   20.0.0.1/32 is directly connected, 00:21:33, gigabitethernet1
R   100.0.0.0/24 [120/1] via 20.0.0.2, 00:10:58, gigabitethernet1
R   110.0.0.0/24 [120/1] via 20.0.0.2, 00:10:58, gigabitethernet1
C   127.0.0.0/8 is directly connected, 30:20:17, lo0
L   127.0.0.1/32 is directly connected, 30:20:17, lo0
```

RIP routes have been learnt by Device2.

Step 3: Configure the routing policy.

#Configure Device2.

```
Device2(config)#ip access-list standard 1
Device2(config-std-nacl)#permit 100.0.0.0 0.0.0.255
Device2(config-std-nacl)#exit
Device2(config)#route-map RIPtoOSPF
Device2(config-route-map)#match ip address 1
Device2(config-route-map)#exit
```

The route-map is configured to invoke an ACL to match only 100.0.0.0/24 while filtering out other network segment, such as 20.0.0.0/24 and 110.0.0.0/24.

Step 4: Configure OSPF to redistribute RIP routes and associate a routing policy.

#Configure Device2.

```
Device2(config)#router ospf 100
Device2(config-ospf)#redistribute rip route-map RIPtoOSPF
Device2(config-ospf)#exit
```

In redistributing RIP routes, the route-map matching rule is invoked for filtration.

Step 5: Check the result.

#Query the route table of Device1.

```
Device1#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external


C   10.0.0.0/24 is directly connected, 00:47:27, gigabitethernet0
L   10.0.0.1/32 is directly connected, 00:47:27, gigabitethernet0
OE  100.0.0.0/24 [150/20] via 10.0.0.2, 00:21:39, gigabitethernet0
C   127.0.0.0/8 is directly connected, 21:40:06, lo0
L   127.0.0.1/32 is directly connected, 21:40:06, lo0
```

The route table of Device1 has learnt only the OSPF external route 100.0.0.0/24 while routes 20.0.0.0/24 and 110.0.0.0/24 have been filtered out.

#Query the OSPF process information and database of Device2.

```
Device2#show ip ospf 100
 Routing Process "ospf 100" with ID 2.2.2.2
 Process bound to VRF default
 Process uptime is 1 minute
 IETF NSF restarter support disabled
 IETF NSF helper support enabled
 Conforms to RFC2328, and RFC1583Compatibility flag is disabled
 Supports only single TOS(TOS0) routes
 Supports opaque LSA
 Supports Graceful Restart
 This router is an ASBR (injecting external routing information)
 Initial SPF schedule delay 5000 msecs
 Minimum hold time between two consecutive SPFs 10000 msecs
 Maximum wait time between two consecutive SPFs 10000 msecs
 ASE route calculate executed 0 times Refresh timer 10 secs
 Transmit timer 100 msecs, burst num 1 Number of external LSA 2. Checksum Sum
0x42C812
 Number of opaque AS LSA 0. Checksum Sum 0x000000
 Number of non-default external LSA is 2
 External LSA database is unlimited.
 Not Support Demand Circuit lsa number is 0, autonomy system support flood
DoNotAge Lsa
 Number of areas attached to this router: 1
    Area 0 (BACKBONE)       Number of interfaces in this area is 1(1)
        Number of fully adjacent neighbors in this area is 1
        Number of fully adjacent sham-link neighbors in this area is 0
        Area has no authentication
        SPF algorithm last executed 00:00:15.366 ago
        SPF algorithm executed 3 times
        IA route calculat executed 0 times          Number of LSA 3. Checksum Sum
0x01032d
        Not Support Demand Circuit lsa number is 0,
        Indication lsa (by other routers) number is: 0,
        Area support flood DoNotAge Lsa

Device2#show ip ospf 100 database

            OSPF Router with ID (2.2.2.2) (Process ID 100)

                Router Link States (Area 0)

Link ID       ADV Router       Age Seq#       CkSum  Link count
1.1.1.1       1.1.1.1          191 0x80000004 0x70a0 1
2.2.2.2       2.2.2.2          537 0x80000005 0x36ce 1

                Net Link States (Area 0)

Link ID       ADV Router       Age Seq#       CkSum
10.0.0.2      2.2.2.2          818 0x80000003 0x3fd8

                AS External Link States

Link ID       ADV Router       Age Seq#       CkSum  Route
```

```
100.0.0.0       2.2.2.2         718 0x80000002 0x72be E2 100.0.0.0/24  [0x0]
```

According to the information about OSPF process 100, Device2 has changed its role to become an ASBR, and only one external LSA has been generated in the database.

Note:

- In an actual application, if there are two or more AS boundary routers, it is recommended that you do not redistribute routes between different routing protocols. If you really need to redistribute routes between different routing protocols, configure a routing policy to prevent routing loops.

6.3.4. Configure OSPF Multi-Processes

Network Requirements

- Configure OSPF on all devices. On Device2, enable two OSPF processes. Configure OSPF 100 of Device1 and that of Device2 to set up a neighbor relation. Configure OSPF 200 of Device3 and that of Device2 to set up a neighbor relation.
- The two OSPF processes on Device2 redistribute routes to each other. OSPF process 100 uses a routing policy to control to redistribute only route 110.0.0.0/24. OSPF process 200 uses a routing policy to control to redistribute only route 100.0.0.0/24.

Network Topology

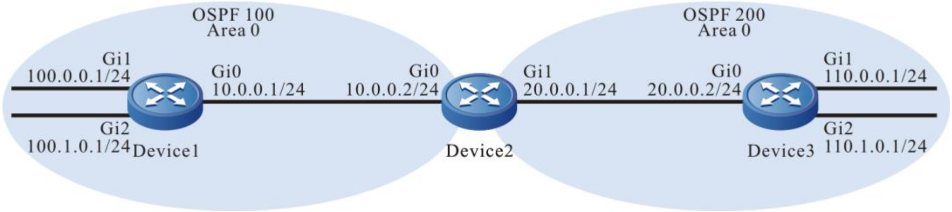

Figure 6-4 Networking for configuring OSPF multi-processes

Configuration Steps

Step 1: Configure the IP addresses of the interfaces. (Omitted)

Step 2: Configure the OSPF protocol.

#Configure Device1.

```
Device1#configure terminal
Device1(config)#router ospf 100
Device1(config-ospf)#router-id 1.1.1.1
Device1(config-ospf)#network 10.0.0.0 0.0.0.255 area 0
Device1(config-ospf)#network 100.0.0.0 0.0.0.255 area 0
Device1(config-ospf)#network 100.1.0.0 0.0.0.255 area 0
Device1(config-ospf)#exit
```

#On Device2, create two OSPF processes, process 100 and process 200.

```
Device2#configure terminal
Device2(config)#router ospf 100
Device2(config-ospf)#router-id 2.2.2.2
Device2(config-ospf)#network 10.0.0.0 0.0.0.255 area 0
```

```
Device2(config-ospf)#exit
Device2(config)#router ospf 200
Device2(config-ospf)#router-id 2.2.2.3
Device2(config-ospf)#network 20.0.0.0 0.0.0.255 area 0
Device2(config-ospf)#exit
```

#Configure Device3.

```
Device3#configure terminal
Device3(config)#router ospf 200
Device3(config-ospf)#router-id 3.3.3.3
Device3(config-ospf)#network 20.0.0.0 0.0.0.255 area 0
Device3(config-ospf)#network 110.0.0.0 0.0.0.255 area 0
Device3(config-ospf)#network 110.1.0.0 0.0.0.255 area 0
Device3(config-ospf)#exit
```

Note:

- If there exist multiple OSPF processes, it is recommended that you configure different Router IDs for the OSPF processes to prevent Router ID conflict.

#Query the LSDB and neighbor information of Device2.

```
Device2#show ip ospf neighbor
OSPF process 100:
Neighbor ID     Pri   State          Dead Time   Address        Interface
1.1.1.1           1   Full/Backup    00:00:30    10.0.0.1       gigabitethernet0
OSPF process 200:
Neighbor ID     Pri   State          Dead Time   Address        Interface
3.3.3.3           1   Full/DR        00:00:33    20.0.0.2       gigabitethernet1

Device2#show ip ospf database

            OSPF Router with ID (2.2.2.2) (Process ID 100)

                Router Link States (Area 0)

Link ID         ADV Router      Age Seq#       CkSum  Link count
1.1.1.1         1.1.1.1         19 0x80000016 0x53bf 3
2.2.2.2         2.2.2.2         15 0x80000010 0x1ae1 1

                Net Link States (Area 0)
```

```
Link ID       ADV Router     Age Seq#     CkSum
10.0.0.2      2.2.2.2        21 0x80000001 0x43d6

              OSPF Router with ID (2.2.2.3) (Process ID 200)

                Router Link States (Area 0)

Link ID       ADV Router     Age Seq#     CkSum  Link count
2.2.2.3       2.2.2.3        14 0x8000000f 0xb235 1
3.3.3.3       3.3.3.3        15 0x8000001b 0x696b 3

                Net Link States (Area 0)

Link ID       ADV Router     Age Seq#     CkSum
20.0.0.2      3.3.3.3        15 0x80000002 0x03fe
```

Neighbors have been set up respectively for OSPF 100 and OSPF 200 of Device2, and the two processes have their respective OSPF databases.

#Query the OSPF route table of Device2.

```
Device2#show ip ospf route
OSPF process 100:
Codes: C - connected, D - Discard, O - OSPF, IA - OSPF inter area
    N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
    E1 - OSPF external type 1, E2 - OSPF external type 2

O  10.0.0.0/24 [1] is directly connected, gigabitethernet0, Area 0
O  100.0.0.0/24 [2] via 10.0.0.1, gigabitethernet0, Area 0
O  100.1.0.0/24 [2] via 10.0.0.1, gigabitethernet0, Area 0
OSPF process 200:
Codes: C - connected, D - Discard, O - OSPF, IA - OSPF inter area
    N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
    E1 - OSPF external type 1, E2 - OSPF external type 2

O  20.0.0.0/24 [1] is directly connected, gigabitethernet1, Area 0
O  110.0.0.0/24 [2] via 20.0.0.2, gigabitethernet1, Area 0
O  110.1.0.0/24 [2] via 20.0.0.2, gigabitethernet1, Area 0
```

OSPF process 100 and process 200 have calculated their own routes.

#Query the route table of Device2.

```
Device2#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
      U - Per-user Static route
      O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

C   10.0.0.0/24 is directly connected, 00:05:34, gigabitethernet0
L   10.0.0.2/32 is directly connected, 00:05:34, gigabitethernet0
C   20.0.0.0/24 is directly connected, 00:05:28, gigabitethernet1
L   20.0.0.1/32 is directly connected, 00:05:28, gigabitethernet1
O   100.0.0.0/24 [110/2] via 10.0.0.1, 00:04:42, gigabitethernet0
O   100.1.0.0/24 [110/2] via 10.0.0.1, 00:04:42, gigabitethernet0
O   110.0.0.0/24 [110/2] via 20.0.0.2, 00:04:41, gigabitethernet1
O   110.1.0.0/24 [110/2] via 20.0.0.2, 00:04:41, gigabitethernet1
C   127.0.0.0/8 is directly connected, 48:40:33, lo0
L   127.0.0.1/32 is directly connected, 48:40:33, lo0
```

Step 3: Configure the routing policy.

#Configure Device2.

```
Device2(config)#ip prefix-list 1 permit 110.0.0.0/24
Device2(config)#ip prefix-list 2 permit 100.0.0.0/24
Device2(config)#route-map OSPF200to100
Device2(config-route-map)#match ip address prefix-list 1
Device2(config-route-map)#exit
Device2(config)#route-map OSPF100to200
Device2(config-route-map)#match ip address prefix-list 2
Device2(config-route-map)#exit
```

Note:

- The route-maps have been configured to invoke prefix list 1 and prefix list 2 to match network segment 110.0.0.0/24 and 100.0.0.0/24 respectively.
- In configuring a routing policy, you can create a filtration rule based on a prefix list or ACL. The prefix list can precisely match routing masks while the ACL cannot match routing masks.

Step 4: Configure OSPF processes to redistribute RIP routes to each other and associate routing policies.

#Configure Device2.

```
Device2(config)#router ospf 100
Device2(config-ospf)#redistribute ospf 200 route-map OSPF200to100
Device2(config-ospf)#exit
```

```
Device2(config)#router ospf 200
Device2(config-ospf)#redistribute ospf 100 route-map OSPF100to200
Device2(config-ospf)#exit
```

Step 5: Check the result.

#Query OSPF LSDB of Device2.

```
Device2#show ip ospf database

            OSPF Router with ID (2.2.2.2) (Process ID 100)

                Router Link States (Area 0)

Link ID         ADV Router      Age Seq#        CkSum  Link count
1.1.1.1         1.1.1.1         1663 0x80000016 0x53bf 3
2.2.2.2         2.2.2.2         216 0x80000011 0x1eda 1

                Net Link States (Area 0)

Link ID         ADV Router      Age Seq#        CkSum
10.0.0.2        2.2.2.2         1664 0x80000001 0x43d6

                AS External Link States

Link ID         ADV Router      Age Seq#        CkSum  Route
110.0.0.0       2.2.2.2         216 0x80000001 0x3dfc E2 110.0.0.0/24  [0x0]

            OSPF Router with ID (2.2.2.3) (Process ID 200)

                Router Link States (Area 0)

Link ID         ADV Router      Age Seq#        CkSum  Link count
2.2.2.3         2.2.2.3         205 0x80000010 0xb62e 1
3.3.3.3         3.3.3.3         1658 0x8000001b 0x696b 3

                Net Link States (Area 0)

Link ID         ADV Router      Age Seq#        CkSum
```

```
20.0.0.2       3.3.3.3        1658 0x80000002 0x03fe


                AS External Link States


Link ID        ADV Router      Age Seq#      CkSum  Route
100.0.0.0      2.2.2.3         205 0x80000001 0xb989 E2 100.0.0.0/24  [0x0]
```

According to the queried information, OSPF process 100 has only the LSA of external route 110.0.0.0/24, and the other routes 110.1.0.0/24 and 20.0.0.0/24 have been filtered out by the routing policy OSPF200to100. Similarly, OSPF process 200 has only the LSA of external route 100.0.0.0/24, and the other routes 100.1.0.0/24 and 10.0.0.0/24 have been filtered out by the routing policy OSPF100to200.

#Query the route table of Device1.

```
Device1#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external


C   10.0.0.0/24 is directly connected, 00:40:20, gigabitethernet0
L   10.0.0.1/32 is directly connected, 00:40:20, gigabitethernet0
C   100.0.0.0/24 is directly connected, 03:11:36, gigabitethernet1
C   100.0.0.1/32 is directly connected, 03:11:36, gigabitethernet1
C   100.1.0.0/24 is directly connected, 01:00:22, gigabitethernet2
L   100.1.0.1/32 is directly connected, 01:00:22, gigabitethernet2
OE  110.0.0.0/24 [150/2] via 10.0.0.2, 00:15:27, gigabitethernet0
C   127.0.0.0/8 is directly connected, 97:08:23, lo0
L   127.0.0.1/32 is directly connected, 97:08:23, lo0
```

Device1 has only learnt route 110.0.0.0/24.

#Query the route table of Device3.

```
Device3#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external


C   20.0.0.0/24 is directly connected, 00:42:44, gigabitethernet0
L   20.0.0.2/32 is directly connected, 00:42:44, gigabitethernet0
OE  100.0.0.0/24 [150/2] via 20.0.0.1, 00:17:45, gigabitethernet0
C   110.0.0.0/24 is directly connected, 01:02:03, gigabitethernet1
L   110.0.0.1/32 is directly connected, 01:02:03, gigabitethernet1
```

```
C   110.1.0.0/24 is directly connected, 01:02:14, gigabitethernet2
L   110.1.0.1/32 is directly connected, 01:02:14, gigabitethernet2
C   127.0.0.0/8 is directly connected, 41:02:01, lo0
L   127.0.0.1/32 is directly connected, 41:02:01, lo0
```

Device3 has only learnt route 100.0.0.0/24.

Caution:

- In an actual application, if there are two or more AS boundary routers, it is recommended that you do not redistribute routes between different OSPF processes. If you really need to redistribute routes between different OSPF processes, configure a route filtering policy to prevent routing loops.

6.3.5. Configure OSPF External Route Summary

Network Requirements

- Run OSPF between Device1 and Device2, and run RIPv2 between Device2 and Device3.
- Device2 redistributes RIP routes to OSPF. To decrease the number of routes on Device1, summarize the redistributed RIP routes into summary route 20.0.0.0/16.

Network Topology

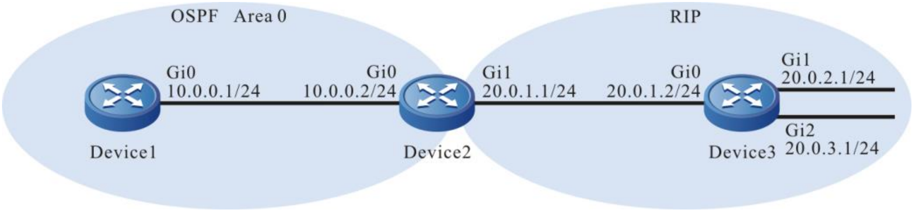

Figure 6-5 Networking for configuring OSPF external route summary

Configuration Steps

Step 1: Configure the IP addresses of the interfaces. (Omitted)

Step 2: #Configure OSPF and RIPv2.

#Configure OSPF for Device1.

```
Device1#configure terminal
Device1(config)#router ospf 100
Device1(config-ospf)#router-id 1.1.1.1
Device1(config-ospf)#network 10.0.0.0 0.0.0.255 area 0
Device1(config-ospf)#exit
```

#Configure OSPF and RIPv2 for Device2.

```
Device2#configure terminal
Device2(config)#router ospf 100
Device2(config-ospf)#router-id 2.2.2.2
Device2(config-ospf)#network 10.0.0.0 0.0.0.255 area 0
Device2(config-ospf)#exit
Device2(config)#router rip
```

```
Device2(config-rip)#version 2
Device2(config-rip)#network 20.0.0.0
Device2(config-rip)#exit
```

#Configure RIPv2 for Device3.

```
Device3#configure terminal
Device3(config)#router rip
Device3(config-rip)#version 2
Device3(config-rip)#network 20.0.0.0
Device3(config-rip)#exit
```

#Query the route table of Device2.

```
Device2#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external
C   10.0.0.0/24 is directly connected, 00:15:46, gigabitethernet0
L   10.0.0.2/32 is directly connected, 00:15:46, gigabitethernet0
C   20.0.1.0/24 is directly connected, 00:15:23, gigabitethernet1
L   20.0.1.1/32 is directly connected, 00:15:23, gigabitethernet1
R   20.0.2.0/24 [120/1] via 20.0.1.2, 00:12:17, gigabitethernet1
R   20.0.3.0/24 [120/1] via 20.0.1.2, 00:12:06, gigabitethernet1
C   127.0.0.0/8 is directly connected, 03:34:27, lo0
L   127.0.0.1/32 is directly connected, 03:34:27, lo0
```

Step 3: Configure OSPF to redistribute RIP routes.

#Configure Device2.

```
Device2(config)#router ospf 100
Device2(config-ospf)#redistribute rip
Device2(config-ospf)#exit
```

#Query OSPF LSDB of Device2.

```
Device2#show ip ospf database

            OSPF Router with ID (2.2.2.2) (Process ID 100)

                Router Link States (Area 0)

Link ID         ADV Router      Age Seq#        CkSum  Link count
```

```
1.1.1.1         1.1.1.1         1071 0x80000003 0x729f 1
2.2.2.2         2.2.2.2         873 0x80000004 0x38cd 1

                Net Link States (Area 0)

Link ID         ADV Router      Age Seq#        CkSum
10.0.0.2        2.2.2.2         1070 0x80000001 0x43d6

                AS External Link States

Link ID         ADV Router      Age Seq#        CkSum  Route
20.0.1.0        2.2.2.2         365 0x80000001 0x7d04 E2 20.0.1.0/24  [0x0]
20.0.2.0        2.2.2.2         365 0x80000001 0x720e E2 20.0.2.0/24  [0x0]
20.0.3.0        2.2.2.2         365 0x80000001 0x6718 E2 20.0.3.0/24  [0x0]
```

According to the OSPF database, three external LSA have been generated, indicating that the RIP routes have been redistributed to OSPF.

#Query the route table of Device1.

```
Device1#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external t

C   10.0.0.0/24 is directly connected, 00:56:40, gigabitethernet0
L   10.0.0.1/32 is directly connected, 00:56:40, gigabitethernet0
OE  20.0.1.0/24 [150/20] via 10.0.0.2, 00:02:40, gigabitethernet0
OE  20.0.2.0/24 [150/20] via 10.0.0.2, 00:02:40, gigabitethernet0
OE  20.0.3.0/24 [150/20] via 10.0.0.2, 00:02:40, gigabitethernet0
C   127.0.0.0/8 is directly connected, 115:12:28, lo0
L   127.0.0.1/32 is directly connected, 115:12:28, lo0
```

Device1 has learnt redistributed RIP routes.

Step 4: On the ASBR, configure OSPF external route summary. Now Device2 is the ASBR.

#Configure Device2 and summarize the redistributed RIP routes into 20.0.0.0/16.

```
Device2(config)#router ospf 100
Device2(config-ospf)#summary-address 20.0.0.0 255.255.0.0
Device2(config-ospf)#exit
```

Step 5: Check the result.

#Query OSPF LSDB of Device2.

```
Device2#show ip ospf database

            OSPF Router with ID (2.2.2.2) (Process ID 100)

                Router Link States (Area 0)

Link ID         ADV Router      Age Seq#        CkSum  Link count
1.1.1.1         1.1.1.1         1437 0x80000003 0x729f 1
2.2.2.2         2.2.2.2         1240 0x80000004 0x38cd 1

                Net Link States (Area 0)

Link ID         ADV Router      Age Seq#        CkSum
10.0.0.2        2.2.2.2         144 0x80000002 0x41d7

                AS External Link States

Link ID         ADV Router      Age Seq#        CkSum  Route
20.0.0.0        2.2.2.2         84 0x80000001 0x88f9 E2 20.0.0.0/16  [0x0]
```

Comparing the result with the result in Step 3, you will find that the three external LSAs have been deleted, and a summarized external LSA has been generated.

#Query the route table of Device2.

```
Device2#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

C   10.0.0.0/24 is directly connected, 00:28:03, gigabitethernet0
L   10.0.0.2/32 is directly connected, 00:28:03, gigabitethernet0
O   20.0.0.0/16 [110/1] is directly connected, 00:04:48, null0
C   20.0.1.0/24 is directly connected, 00:27:40, gigabitethernet1
L   20.0.1.1/32 is directly connected, 00:27:40, gigabitethernet1
R   20.0.2.0/24 [120/1] via 20.0.1.2, 00:24:34, gigabitethernet1
R   20.0.3.0/24 [120/1] via 20.0.1.2, 00:24:23, gigabitethernet1
C   127.0.0.0/8 is directly connected, 03:46:44, lo0
```

```
L   127.0.0.1/32 is directly connected, 03:46:44, lo0
```

Note:

- In the route table of Device2, a summary route 20.0.0.0/16 with the output interface being Null0 has been automatically added. This route helps to prevent routing loops.

#Query the route table of Device1.

```
Device1#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

C   10.0.0.0/24 is directly connected, 00:58:40, gigabitethernet0
L   10.0.0.1/32 is directly connected, 00:58:40, gigabitethernet0
OE  20.0.0.0/16 [150/20] via 10.0.0.2, 00:15:26, gigabitethernet0
C   127.0.0.0/8 is directly connected, 115:17:28, lo0
L   127.0.0.1/32 is directly connected, 115:17:28, lo0
```

The route table of Device1 has only learnt the summary route 20.0.0.0/16.

6.3.6. Configure Route Summary on Inter-Area OSPF Routes

Network Requirements

- Configure OSPF for all devices, and divide the devices into two areas, Area 0 and Area 1.
- To decrease the number of inter-area routes, inter-area routes are summarized on the ABR. The routes in Area 0 are summarized to form 10.0.0.0/16. The routes in Area 1 are summarized to form 20.0.0.0/16.

Network Topology

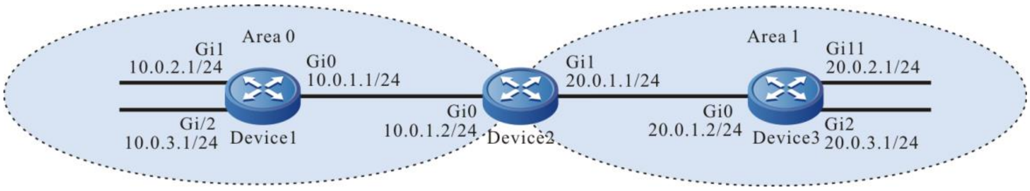

Figure 6-6 Networking for configuring route summary on inter-area OSPF routes

Configuration Steps

Step 1: Configure the IP addresses of the interfaces. (Omitted)

Step 2: Configure an OSPF process and let the interface cover different areas.

#Configure Device1.

```
Device1#configure terminal
Device1(config)#router ospf 100
Device1(config-ospf)#router-id 1.1.1.1
Device1(config-ospf)#network 10.0.1.0 0.0.0.255 area 0
Device1(config-ospf)#network 10.0.2.0 0.0.0.255 area 0
```

```
Device1(config-ospf)#network 10.0.3.0 0.0.0.255 area 0
Device1(config-ospf)#exit
```

#Configure Device2.

```
Device2#configure terminal
Device2(config)#router ospf 100
Device2(config-ospf)#router-id 2.2.2.2
Device2(config-ospf)#network 10.0.1.0 0.0.0.255 area 0
Device2(config-ospf)#network 20.0.1.0 0.0.0.255 area 1
Device2(config-ospf)#exit
```

#Configure Device3.

```
Device3#configure terminal
Device3(config)#router ospf 100
Device3(config-ospf)#router-id 3.3.3.3
Device3(config-ospf)#network 20.0.1.0 0.0.0.255 area 1
Device3(config-ospf)#network 20.0.2.0 0.0.0.255 area 1
Device3(config-ospf)#network 20.0.3.0 0.0.0.255 area 1
Device3(config-ospf)#exit
```

#Query the OSPF LSDB and route table of Device2.

```
Device2#show ip ospf database

            OSPF Router with ID (2.2.2.2) (Process ID 100)

                Router Link States (Area 0)

Link ID         ADV Router      Age Seq#       CkSum  Link count
1.1.1.1         1.1.1.1         1419 0x80000007 0x4f81 3
2.2.2.2         2.2.2.2         1414 0x80000004 0x4bb9 1

                Net Link States (Area 0)

Link ID         ADV Router      Age Seq#       CkSum
10.0.1.2        2.2.2.2         1419 0x80000001 0x38e0

                Summary Link States (Area 0)

Link ID         ADV Router      Age Seq#       CkSum  Route
20.0.1.0        2.2.2.2         1437 0x80000001 0x47d7 20.0.1.0/24
```

```
20.0.2.0        2.2.2.2        1363 0x80000001 0x46d6 20.0.2.0/24
20.0.3.0        2.2.2.2        1363 0x80000001 0x3be0 20.0.3.0/24

                Router Link States (Area 1)

Link ID         ADV Router      Age Seq#        CkSum  Link count
2.2.2.2        2.2.2.2         1368 0x80000004 0xe70b 1
3.3.3.3        3.3.3.3         1341 0x80000006 0x6138 3

                Net Link States (Area 1)

Link ID         ADV Router      Age Seq#        CkSum
20.0.1.1       2.2.2.2         1368 0x80000001 0x24e3

                Summary Link States (Area 1)

Link ID         ADV Router      Age Seq#        CkSum  Route
10.0.1.0       2.2.2.2         1442 0x80000001 0xc95f 10.0.1.0/24
10.0.2.0        2.2.2.2         1409 0x80000001 0xc85e 10.0.2.0/24
10.0.3.0        2.2.2.2         1409 0x80000001 0xbd68 10.0.3.0/24

Device2#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

C   10.0.1.0/24 is directly connected, 00:30:31, gigabitethernet0
L   10.0.1.2/32 is directly connected, 00:30:31, gigabitethernet0
O   10.0.2.0/24 [110/2] via 10.0.1.1, 00:23:37, gigabitethernet0
O   10.0.3.0/24 [110/2] via 10.0.1.1, 00:23:37, gigabitethernet0
C   20.0.1.0/24 is directly connected, 02:09:10, gigabitethernet1
L   20.0.1.1/32 is directly connected, 02:09:10, gigabitethernet1
O   20.0.2.0/24 [110/2] via 20.0.1.2, 00:22:51, gigabitethernet1
O   20.0.3.0/24 [110/2] via 20.0.1.2, 00:22:51, gigabitethernet1
C   127.0.0.0/8 is directly connected, 05:28:14, lo0
L   127.0.0.1/32 is directly connected, 05:28:14, lo0
```

In the OSPF database of Device2, three inter-area LSAs are generated respectively for Area 0 and Area 1. The intra-area routes of the areas have also been added into the route table.

#Query the OSPF LSDB and route table of Device1.

```
Device1#show ip ospf database

      OSPF Router with ID (1.1.1.1) (Process ID 100)

        Router Link States (Area 0)

Link ID       ADV Router      Age Seq#      CkSum  Link count
1.1.1.1       1.1.1.1         249 0x80000008 0x4d82 3
2.2.2.2       2.2.2.2         191 0x80000005 0x49ba 1

        Net Link States (Area 0)

Link ID       ADV Router      Age Seq#      CkSum
10.0.1.2      2.2.2.2         471 0x80000002 0x36e1

        Summary Link States (Area 0)

Link ID       ADV Router      Age Seq#      CkSum  Route
20.0.1.0      2.2.2.2         251 0x80000002 0x45d8 20.0.1.0/24
20.0.2.0      2.2.2.2         1988 0x80000001 0x46d6 20.0.2.0/24
20.0.3.0      2.2.2.2         1988 0x80000001 0x3be0 20.0.3.0/24

Device1#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
    U - Per-user Static route
    O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

C  10.0.1.0/24 is directly connected, 00:25:11, gigabitethernet0
L  10.0.1.1/32 is directly connected, 00:25:11, gigabitethernet0
C  10.0.2.0/24 is directly connected, 00:24:58, gigabitethernet1
L  10.0.2.1/32 is directly connected, 00:24:58, gigabitethernet1
C  10.0.3.0/24 is directly connected, 00:24:44, gigabitethernet2
L  10.0.3.1/32 is directly connected, 00:24:44, gigabitethernet2
O  20.0.1.0/24 [110/2] via 10.0.1.2, 00:14:59, gigabitethernet0
```

```
O   20.0.2.0/24 [110/3] via 10.0.1.2, 00:14:12, gigabitethernet0
O   20.0.3.0/24 [110/3] via 10.0.1.2, 00:14:12, gigabitethernet0
C   127.0.0.0/8 is directly connected, 116:19:42, lo0
L   127.0.0.1/32 is directly connected, 116:19:42, lo0
```

The OSPF database of Device1 contains three inter-area LSAs, and the three routes have been added into the route table.

#Query the OSPF LSDB and route table of Device3.

```
Device3#show ip ospf database

            OSPF Router with ID (3.3.3.3) (Process ID 100)

                Router Link States (Area 1)

Link ID         ADV Router      Age Seq#        CkSum  Link count
2.2.2.2         2.2.2.2         532 0x80000005 0xe50c 1
3.3.3.3         3.3.3.3         506 0x80000007 0x5f39 3

                Net Link States (Area 1)

Link ID         ADV Router      Age Seq#        CkSum
20.0.1.1        2.2.2.2         532 0x80000002 0x22e4

                Summary Link States (Area 1)

Link ID         ADV Router      Age Seq#        CkSum  Route
10.0.1.0        2.2.2.2          82 0x80000002 0xc760 10.0.1.0/24
10.0.2.0        2.2.2.2         382 0x80000002 0xc65f 10.0.2.0/24
10.0.3.0        2.2.2.2         262 0x80000002 0xbb69 10.0.3.0/24

Device3#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

O   10.0.1.0/24 [110/2] via 20.0.1.1, 00:24:04, gigabitethernet0
O   10.0.2.0/24 [110/3] via 20.0.1.1, 00:24:04, gigabitethernet0
O   10.0.3.0/24 [110/3] via 20.0.1.1, 00:24:04, gigabitethernet0
```

```
C   20.0.1.0/24 is directly connected, 02:09:51, gigabitethernet0
L   20.0.1.2/32 is directly connected, 02:09:51, gigabitethernet0
C   20.0.2.0/24 is directly connected, 02:07:21, gigabitethernet1
L   20.0.2.1/32 is directly connected, 02:07:21, gigabitethernet1
C   20.0.3.0/24 is directly connected, 02:07:09, gigabitethernet2
L   20.0.3.1/32 is directly connected, 02:07:09, gigabitethernet2
C   127.0.0.0/8 is directly connected, 360:20:45, lo0
L   127.0.0.1/32 is directly connected, 360:20:45, lo0
```

Similarly, the OSPF database of Device3 contains three inter-area LSAs, and the three routes have been added into the route table.

Step 3: On the ABR, configure inter-area route summary. Now Device2 is the ABR.

#On Device2, summarize the routes in Area 0 to form route 10.0.0.0/16, and summarize the routes in Area 1 to form route 20.0.0.0/16.

```
Device2(config)#router ospf 100
Device2(config-ospf)#area 0 range 10.0.0.0/16
Device2(config-ospf)#area 1 range 20.0.0.0/16
Device2(config-ospf)#exit
```

Step 4: Check the result.

#Query the OSPF LSDB and route table of Device2.

```
Device2#show ip ospf database

            OSPF Router with ID (2.2.2.2) (Process ID 100)

                Router Link States (Area 0)

Link ID        ADV Router       Age Seq#        CkSum  Link count
1.1.1.1        1.1.1.1          305 0x80000009 0x4b83 3
2.2.2.2        2.2.2.2          297 0x80000006 0x47bb 1

                Net Link States (Area 0)

Link ID        ADV Router       Age Seq#        CkSum
10.0.1.2       2.2.2.2          527 0x80000003 0x34e2

                Summary Link States (Area 0)

Link ID        ADV Router       Age Seq#        CkSum  Route
```

```
20.0.0.0        2.2.2.2          23 0x80000001 0x52cd 20.0.0.0/16

                Router Link States (Area 1)

Link ID         ADV Router       Age Seq#       CkSum  Link count
2.2.2.2         2.2.2.2          277 0x80000006 0xe30d 1
3.3.3.3         3.3.3.3          332 0x80000008 0x5d3a 3

                Net Link States (Area 1)

Link ID         ADV Router       Age Seq#       CkSum
20.0.1.1        2.2.2.2          317 0x80000003 0x20e5

                Summary Link States (Area 1)

Link ID         ADV Router       Age Seq#       CkSum  Route
10.0.0.0        2.2.2.2          26 0x80000001 0xd455 10.0.0.0/16

Device2#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

O   10.0.0.0/16 [110/1] is directly connected, 00:00:31, null0
C   10.0.1.0/24 is directly connected, 00:40:31, gigabitethernet0
L   10.0.1.2/32 is directly connected, 00:40:31, gigabitethernet0
O   10.0.2.0/24 [110/2] via 10.0.1.1, 00:33:37, gigabitethernet0
O   10.0.3.0/24 [110/2] via 10.0.1.1, 00:33:37, gigabitethernet0
O   20.0.0.0/16 [110/1] is directly connected, 00:00:27, null0
C   20.0.1.0/24 is directly connected, 02:19:10, gigabitethernet1
L   20.0.1.1/32 is directly connected, 02:19:10, gigabitethernet1
O   20.0.2.0/24 [110/2] via 20.0.1.2, 00:32:51, gigabitethernet1
O   20.0.3.0/24 [110/2] via 20.0.1.2, 00:32:51, gigabitethernet1
C   127.0.0.0/8 is directly connected, 05:38:14, lo0
L   127.0.0.1/32 is directly connected, 05:38:14, lo0
```

Comparing the result with the result of Step 2, you will find that only one summarized inter-area LSA is generated respectively for Area 0 and Area 1 in the OSPF database of Device2. Similarly, a summary route with the output interface being Null0 is automatically added into the route table.

#Query the OSPF LSDB and route table of Device1.

```
Device1#show ip ospf database

            OSPF Router with ID (1.1.1.1) (Process ID 100)

                Router Link States (Area 0)

Link ID         ADV Router      Age Seq#       CkSum  Link count
1.1.1.1         1.1.1.1         1338 0x80000009 0x4b83 3
2.2.2.2         2.2.2.2         1332 0x80000006 0x47bb 1

                Net Link States (Area 0)

Link ID         ADV Router      Age Seq#       CkSum
10.0.1.2        2.2.2.2         1563 0x80000003 0x34e2

                Summary Link States (Area 0)

Link ID         ADV Router      Age Seq#       CkSum  Route
20.0.0.0        2.2.2.2           90 0x80000001 0x52cd 20.0.0.0/16

Device1#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

C   10.0.1.0/24 is directly connected, 00:40:11, gigabitethernet0
L   10.0.1.1/32 is directly connected, 00:40:11, gigabitethernet0
C   10.0.2.0/24 is directly connected, 00:39:58, gigabitethernet1
L   10.0.2.1/32 is directly connected, 00:39:58, gigabitethernet1
C   10.0.3.0/24 is directly connected, 00:39:44, gigabitethernet2
L   10.0.3.1/32 is directly connected, 00:39:44, gigabitethernet2
O   20.0.0.0/16 [110/2] via 10.0.1.2, 00:02:18, gigabitethernet0
C   127.0.0.0/8 is directly connected, 116:44:42, lo0
L   127.0.0.1/32 is directly connected, 116:44:42, lo0
```

On Device1, you will find that the OSPF database contains only the summarized inter-area LSA, and the route table leans only the summary route 20.0.0.0/16 of Area 1. Similarly, Device3 learns only the summary route 10.0.0.0/16 of Area 0.

6.3.7. Configure Route Filtering on Inter-Area OSPF Routes

Network Requirements

- Configure OSPF for all devices, and divide the devices into two areas, Area 0 and Area 1.
- On the ABR, configure inter-area route filtration. According to route filtration, Area 0 does not allow injection of route 20.0.3.0/24, and 10.0.3.0/24 is not allowed to flood into other areas.

Network Topology

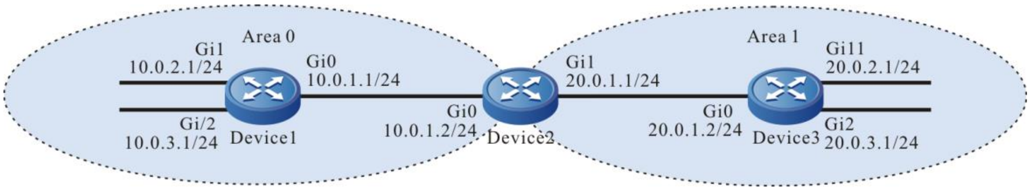

Figure 6-7 Networking for configuring route filtering on inter-area OSPF routes

Configuration Steps

Step 1: Configure the IP addresses of the interfaces. (Omitted)

Step 2: Configure an OSPF process and let the interface cover different areas.

#Configure Device1.

```
Device1#configure terminal
Device1(config)#router ospf 100
Device1(config-ospf)#router-id 1.1.1.1
Device1(config-ospf)#network 10.0.1.0 0.0.0.255 area 0
Device1(config-ospf)#network 10.0.2.0 0.0.0.255 area 0
Device1(config-ospf)#network 10.0.3.0 0.0.0.255 area 0
Device1(config-ospf)#exit
```

#Configure Device2.

```
Device2#configure terminal
Device2(config)#router ospf 100
Device2(config-ospf)#router-id 2.2.2.2
Device2(config-ospf)#network 10.0.1.0 0.0.0.255 area 0
Device2(config-ospf)#network 20.0.1.0 0.0.0.255 area 1
Device2(config-ospf)#exit
```

#Configure Device3.

```
Device3#configure terminal
Device3(config)#router ospf 100
Device3(config-ospf)#router-id 3.3.3.3
Device3(config-ospf)#network 20.0.1.0 0.0.0.255 area 1
Device3(config-ospf)#network 20.0.2.0 0.0.0.255 area 1
```

```
Device3(config-ospf)#network 20.0.3.0 0.0.0.255 area 1
Device3(config-ospf)#exit
```

#Query the OSPF LSDB and route table of Device2.

```
Device2#show ip ospf database

        OSPF Router with ID (2.2.2.2) (Process ID 100)

          Router Link States (Area 0)

Link ID        ADV Router      Age Seq#       CkSum  Link count
1.1.1.1        1.1.1.1         329 0x8000005b 0xa6d5 3
2.2.2.2        2.2.2.2         324 0x80000051 0xb007 1

          Net Link States (Area 0)

Link ID        ADV Router      Age Seq#       CkSum
10.0.1.2       2.2.2.2         324 0x8000004e 0x9d2e

          Summary Link States (Area 0)

Link ID        ADV Router      Age Seq#       CkSum  Route
20.0.1.0       2.2.2.2         324 0x8000004e 0xac25 20.0.1.0/24
20.0.2.0       2.2.2.2         324 0x8000004d 0xad23 20.0.2.0/24
20.0.3.0       2.2.2.2         259 0x80000001 0x3be0 20.0.3.0/24

          Router Link States (Area 1)

Link ID        ADV Router      Age Seq#       CkSum  Link count
2.2.2.2        2.2.2.2         334 0x80000055 0x4f51 1
3.3.3.3        3.3.3.3         335 0x80000059 0xca7a 3

          Net Link States (Area 1)

Link ID        ADV Router      Age Seq#       CkSum
20.0.1.2       3.3.3.3         340 0x80000001 0xeb17

          Summary Link States (Area 1)
```

```
Link ID        ADV Router      Age Seq#       CkSum  Route
10.0.1.0       2.2.2.2          365 0x80000001 0xc95f 10.0.1.0/24
10.0.2.0       2.2.2.2          319 0x80000001 0xc85e 10.0.2.0/24
10.0.3.0       2.2.2.2          256 0x80000001 0xbd68 10.0.3.0/24

Device2#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

C   10.0.1.0/24 is directly connected, 00:06:13, gigabitethernet0
L   10.0.1.2/32 is directly connected, 00:06:13, gigabitethernet0
O   10.0.2.0/24 [110/2] via 10.0.1.1, 00:05:22, gigabitethernet0
O   10.0.3.0/24 [110/2] via 10.0.1.1, 00:05:22, gigabitethernet0
C   20.0.1.0/24 is directly connected, 00:06:19, gigabitethernet1
L   20.0.1.1/32 is directly connected, 00:06:19, gigabitethernet1
O   20.0.2.0/24 [110/2] via 20.0.1.2, 00:05:32, gigabitethernet1
O   20.0.3.0/24 [110/2] via 20.0.1.2, 00:05:32, gigabitethernet1
C   127.0.0.0/8 is directly connected, 94:42:22, lo0
L   127.0.0.1/32 is directly connected, 94:42:22, lo0
```

In the OSPF database of Device2, three inter-area LSAs are generated respectively for Area 0 and Area 1.The intra-area routes of the areas have also been added into the route table.

#Query the route table of Device1.

```
Device1#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

C   10.0.1.0/24 is directly connected, 00:08:41, gigabitethernet0
L   10.0.1.1/32 is directly connected, 00:08:41, gigabitethernet0
C   10.0.2.0/24 is directly connected, 37:59:10, gigabitethernet1
L   10.0.2.1/32 is directly connected, 37:59:10, gigabitethernet1
C   10.0.3.0/24 is directly connected, 38:05:36, gigabitethernet2
C   10.0.3.1/32 is directly connected, 38:05:36, gigabitethernet2
O   20.0.1.0/24 [110/2] via 10.0.1.2, 00:07:55, gigabitethernet0
O   20.0.2.0/24 [110/3] via 10.0.1.2, 00:07:55, gigabitethernet0
```

```
O   20.0.3.0/24 [110/3] via 10.0.1.2, 00:06:50, gigabitethernet0
C   127.0.0.0/8 is directly connected, 70:07:32, lo0
L   127.0.0.1/32 is directly connected, 70:07:32, lo0
```

Device1 has learnt routes of Area 1.

#Query the route table of Device3.

```
Device3#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
     U - Per-user Static route
     O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

O   10.0.1.0/24 [110/2] via 20.0.1.1, 00:08:44, gigabitethernet0
O   10.0.2.0/24 [110/3] via 20.0.1.1, 00:08:33, gigabitethernet0
O   10.0.3.0/24 [110/3] via 20.0.1.1, 00:07:30, gigabitethernet0
C   20.0.1.0/24 is directly connected, 00:09:31, gigabitethernet0
L   20.0.1.2/32 is directly connected, 00:09:31, gigabitethernet0
C   20.0.2.0/24 is directly connected, 37:59:57, gigabitethernet1
L   20.0.2.1/32 is directly connected, 37:59:57, gigabitethernet1
C   20.0.3.0/24 is directly connected, 38:03:35, gigabitethernet2
L   20.0.3.1/32 is directly connected, 38:03:35, gigabitethernet2
C   127.0.0.0/8 is directly connected, 61:26:38, lo0
L   127.0.0.1/32 is directly connected, 61:26:38, lo0
```

Device3 has learnt routes of Area 0.

Step 3: Configure a route filtering policy.

#Configure Device2.

```
Device2(config)#ip prefix-list 1 deny 10.0.3.0/24
Device2(config)#ip prefix-list 1 permit 0.0.0.0/0 le 32
Device2(config)#ip prefix-list 2 deny 20.0.3.0/24
Device2(config)#ip prefix-list 2 permit 0.0.0.0/0 le 32
Device2(config)#exit
```

Prefix list 1 filters out network 10.0.3.0/24 and allows all other networks. Prefix list 2 filters out network 20.0.3.0/24 and allows all other networks.

Step 4: On the ABR, configure filtration of inter-area routes and invoke the matching rules of a prefix list.

#Configure Device2.

```
Device2(config)#router ospf 100
Device2(config-ospf)#area 0 filter-list prefix 1 out
Device2(config-ospf)#area 0 filter-list prefix 2 in
```

```
Device2(config-ospf)#exit
```

Step 5: Check the result.

#Query OSPF LSDB of Device2.

```
Device2#show ip ospf database

            OSPF Router with ID (2.2.2.2) (Process ID 100)

                Router Link States (Area 0)

Link ID         ADV Router      Age Seq#        CkSum  Link count
1.1.1.1         1.1.1.1         679 0x8000005b 0xa6d5 3
2.2.2.2         2.2.2.2         673 0x80000051 0xb007 1

                Net Link States (Area 0)

Link ID         ADV Router      Age Seq#        CkSum
10.0.1.2        2.2.2.2         673 0x8000004e 0x9d2e

                Summary Link States (Area 0)

Link ID         ADV Router      Age Seq#        CkSum  Route
20.0.1.0        2.2.2.2         673 0x8000004e 0xac25 20.0.1.0/24
20.0.2.0        2.2.2.2         673 0x8000004d 0xad23 20.0.2.0/24

                Router Link States (Area 1)

Link ID         ADV Router      Age Seq#        CkSum  Link count
2.2.2.2         2.2.2.2         683 0x80000055 0x4f51 1
3.3.3.3         3.3.3.3         684 0x80000059 0xca7a 3

                Net Link States (Area 1)

Link ID         ADV Router      Age Seq#        CkSum
20.0.1.2        3.3.3.3         689 0x80000001 0xeb17

                Summary Link States (Area 1)
```

```
Link ID       ADV Router      Age Seq#       CkSum  Route
10.0.1.0       2.2.2.2          714 0x80000001 0xc95f 10.0.1.0/24
10.0.2.0       2.2.2.2          668 0x80000001 0xc85e 10.0.2.0/24
```

Comparing the result with the result of Step 2, the LSA of network 20.0.3.0/24 has been deleted from Area 0 in the OSPF database. Similarly, the LSA of network 10.0.3.0/24 has been deleted from Area 1.

#Query the route table of Device1.

```
Device1#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

C   10.0.1.0/24 is directly connected, 00:12:57, gigabitethernet0
L   10.0.1.1/32 is directly connected, 00:12:57, gigabitethernet0
C   10.0.2.0/24 is directly connected, 38:03:25, gigabitethernet1
L   10.0.2.1/32 is directly connected, 38:03:25, gigabitethernet1
C   10.0.3.0/24 is directly connected, 38:09:52, gigabitethernet2
L   10.0.3.1/32 is directly connected, 38:09:52, gigabitethernet2
O   20.0.1.0/24 [110/2] via 10.0.1.2, 00:12:11, gigabitethernet0
O   20.0.2.0/24 [110/3] via 10.0.1.2, 00:12:11, gigabitethernet0
C   127.0.0.0/8 is directly connected, 70:11:48, lo0
L   127.0.0.1/32 is directly connected, 70:11:48, lo0
```

The route 20.0.3.0/24 does not exist in the route table of Device1.

#Query the route table of Device3.

```
Device3#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

O   10.0.1.0/24 [110/2] via 20.0.1.1, 00:13:09, gigabitethernet0
O   10.0.2.0/24 [110/3] via 20.0.1.1, 00:12:58, gigabitethernet0
C   20.0.1.0/24 is directly connected, 00:13:56, gigabitethernet0
L   20.0.1.2/32 is directly connected, 00:13:56, gigabitethernet0
C   20.0.2.0/24 is directly connected, 38:04:22, gigabitethernet1
L   20.0.2.1/32 is directly connected, 38:04:22, gigabitethernet1
C   20.0.3.0/24 is directly connected, 38:08:00, gigabitethernet2
```

```
L   20.0.3.1/32 is directly connected, 38:08:00, gigabitethernet2
C   127.0.0.0/8 is directly connected, 64:31:03, lo0
L   127.0.0.1/32 is directly connected, 64:31:03, lo0
```

The route 10.0.3.0/24 does not exist in the route table of Device3.

6.3.8. Configure OSPF Totally Stub Area

Network Requirements

- Configure OSPF for all devices, and divide the devices into three areas: Area 0, Area 1, and Area 2. Configure Area 1 as a totally Stub area.
- On Device4, redistribute a static route to OSPF. After the configuration is completed, the totally Stub area cannot learn inter-area routes and external routes, while the devices of other areas can learn inter-area routes and external routes.

Network Topology

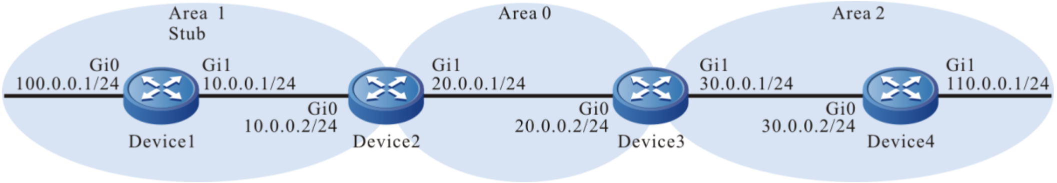

Figure 6-8 Networking for configuring an OSPF totally stub area

Configuration Steps

Step 1: Configure the IP addresses of the interfaces. (Omitted)

Step 2: Configure an OSPF process and let the interfaces cover the related areas.

#Configure Device1. Configure Area 1 to a Stub area.

```
Device1#configure terminal
Device1(config)#router ospf 100
Device1(config-ospf)#router-id 1.1.1.1
Device1(config-ospf)#area 1 stub
Device1(config-ospf)#network 10.0.0.0 0.0.0.255 area 1
Device1(config-ospf)#network 100.0.0.0 0.0.0.255 area 1
Device1(config-ospf)#exit
```

#Configure Device2. Configure Area 1 to a totally Stub area. Devce2 is an ABR, and the **no-summary** command takes effect only on an ABR.

```
Device2#configure terminal
Device2(config)#router ospf 100
Device2(config-ospf)#router-id 2.2.2.2
Device2(config-ospf)#area 1 stub no-summary
Device2(config-ospf)#network 10.0.0.0 0.0.0.255 area 1
Device2(config-ospf)#network 20.0.1.0 0.0.0.255 area 0
Device2(config-ospf)#exit
```

#Configure Device3.

```
Device3#configure terminal
Device3(config)#router ospf 100
Device3(config-ospf)#router-id 3.3.3.3
Device3(config-ospf)#network 20.0.0.0 0.0.0.255 area 0
Device3(config-ospf)#network 30.0.0.0 0.0.0.255 area 2
Device3(config-ospf)#exit
```

#Configure Device4.

```
Device4#configure terminal
Device4(config)#router ospf 100
Device4(config-ospf)#router-id 4.4.4.4
Device4(config-ospf)#network 30.0.0.0 0.0.0.255 area 2
Device4(config-ospf)#network 110.0.0.0 0.0.0.255 area 2
Device4(config-ospf)#exit
```

Step 3: On Device4, configure a static route, and redistribute the route into the OSPF.

#Configure Device4.

```
Device4(config)#ip route 200.1.1.0 255.255.255.0 110.0.0.2
Device4(config)#router ospf 100
Device4(config-ospf)#redistribute static
Device4(config-ospf)#exit
```

Step 4: Check the result.

#Query the OSPF LSDB and route table of Device1.

```
Device1#show ip ospf database

        OSPF Router with ID (1.1.1.1) (Process ID 100)

          Router Link States (Area 1 [Stub])

Link ID       ADV Router      Age Seq#      CkSum  Link count
1.1.1.1       1.1.1.1         19 0x80000009 0x8513 2
2.2.2.2       2.2.2.2         22 0x80000005 0x51b6 1

          Net Link States (Area 1 [Stub])
```

```
Link ID         ADV Router      Age Seq#       CkSum
10.0.0.2        2.2.2.2           22 0x80000001 0x61ba

                Summary Link States (Area 1 [Stub])

Link ID         ADV Router      Age Seq#       CkSum  Route
0.0.0.0         2.2.2.2           55 0x80000002 0x73c1 0.0.0.0/0

Device1#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

O   0.0.0.0/0 [110/2] via 10.0.0.2, 00:00:19, gigabitethernet1
C   10.0.0.0/24 is directly connected, 00:01:04, gigabitethernet1
L   10.0.0.1/32 is directly connected, 00:01:04, gigabitethernet1
C   100.0.0.0/24 is directly connected, 00:11:55, gigabitethernet0
L   100.0.0.1/32 is directly connected, 00:11:55, gigabitethernet0
C   127.0.0.0/8 is directly connected, 30:46:57, lo0
L   127.0.0.1/32 is directly connected, 30:46:57, lo0
```

According to the information in the OSPF database, only Area 1 has an LSA for inter-area route 0.0.0.0/0, while the other areas do not have LSAs for inter-area or external routes. The ABR in the Stub area generates an inter-area route 0.0.0.0/0, which floods in the totally Stub area. Data is forwarded to outside of the area or AS through the default route.

#Query the route table of Device2.

```
Device2#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

C   10.0.0.0/24 is directly connected, 00:01:02, gigabitethernet0
L   10.0.0.2/32 is directly connected, 00:01:02, gigabitethernet0
C   20.0.0.0/24 is directly connected, 00:00:59, gigabitethernet1
L   20.0.0.1/32 is directly connected, 00:00:59, gigabitethernet1
O   30.0.0.0/24 [110/2] via 20.0.0.2, 00:00:17, gigabitethernet1
O   100.0.0.0/24 [110/2] via 10.0.0.1, 00:00:10, gigabitethernet0
O   110.0.0.0/24 [110/3] via 20.0.0.2, 00:00:17, gigabitethernet1
C   127.0.0.0/8 is directly connected, 56:07:04, lo0
```

```
L   127.0.0.1/32 is directly connected, 56:07:04, lo0
OE  200.1.1.0/24 [150/20] via 20.0.0.2, 00:00:16, gigabitethernet1
```

According to the queried information, you will find that Device2 is able to learn inter-area and external routes.

Note:

- If you run the **area** *area-id* **stub** command but do not run the **no-summary** command, the device in the area can learn inter-area routes but cannot learn external routes. Access to a network outside the AS is still conducted through the default route.

6.3.9. Configure OSPF NSSA Area

Network Requirements

- Configure OSPF for all devices, and divide the devices into three areas: Area 0, Area 1, and Area 2. Configure Area 1 and Area 2 as NSSA areas.
- On Device4, redistribute a static route to OSPF. After the configuration is completed, all devices can learn intra-area and inter-area routes, but external routes cannot be injected into Area 1.
- Introduce a default route to the ABR of Area 1 so that Device1 can access an external network through the default route.

Network Topology

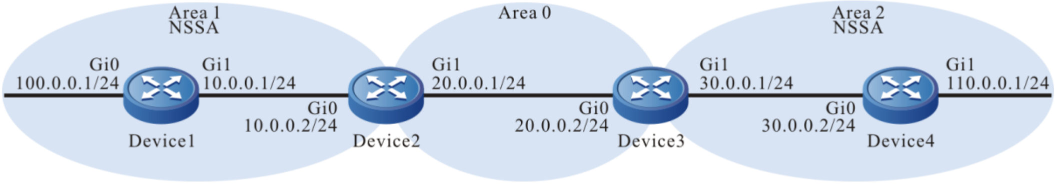

Figure 6-9 Networking for configuring an OSPF NSSA area

Configuration Steps

Step 1: Configure the IP addresses of the interfaces. (Omitted)

Step 2: Configure an OSPF process and let the interfaces cover the related areas.

#Configure Device1. Configure Area 1 to an NSSA area.

```
Device1#configure terminal
Device1(config)#router ospf 100
Device1(config-ospf)#router-id 1.1.1.1
Device1(config-ospf)#area 1 nssa
Device1(config-ospf)#network 10.0.0.0 0.0.0.255 area 1
Device1(config-ospf)#network 100.0.0.0 0.0.0.255 area 1
Device1(config-ospf)#exit
```

#Configure Device2. Configure Area 1 to an NSSA area.

```
Device2#configure terminal
Device2(config)#router ospf 100
Device2(config-ospf)#router-id 2.2.2.2
Device2(config-ospf)#area 1 nssa
```

```
Device2(config-ospf)#network 10.0.0.0 0.0.0.255 area 1
Device2(config-ospf)#network 20.0.1.0 0.0.0.255 area 0
Device2(config-ospf)#exit
```

#Configure Device3. Configure Area 2 to an NSSA area.

```
Device3#configure terminal
Device3(config)#router ospf 100
Device3(config-ospf)#router-id 3.3.3.3
Device3(config-ospf)#area 2 nssa
Device3(config-ospf)#network 20.0.0.0 0.0.0.255 area 0
Device3(config-ospf)#network 30.0.0.0 0.0.0.255 area 2
Device3(config-ospf)#exit
```

#Configure Device4. Configure Area 2 to an NSSA area.

```
Device4#configure terminal
Device4(config)#router ospf 100
Device4(config-ospf)#router-id 4.4.4.4
Device4(config-ospf)#area 2 nssa
Device4(config-ospf)#network 30.0.0.0 0.0.0.255 area 2
Device4(config-ospf)#network 110.0.0.0 0.0.0.255 area 2
Device4(config-ospf)#exit
```

Step 3: On Device4, configure a static route, and redistribute the route into the OSPF.

#Configure Device4.

```
Device4(config)#ip route 200.1.1.0 255.255.255.0 110.0.0.2
Device4(config)#router ospf 100
Device4(config-ospf)#redistribute static
Device4(config-ospf)#exit
```

#Query OSPF LSDB of Device3.

```
Device3#show ip ospf database

            OSPF Router with ID (3.3.3.3) (Process ID 100)

                Router Link States (Area 0)

Link ID         ADV Router      Age Seq#       CkSum  Link count
2.2.2.2         2.2.2.2         179 0x80000004 0xe110 1
3.3.3.3         3.3.3.3         177 0x80000004 0xa345 1
```

```
                Net Link States (Area 0)

Link ID         ADV Router      Age Seq#        CkSum
20.0.0.2        3.3.3.3         182 0x80000001 0xf60d

                Summary Link States (Area 0)

Link ID         ADV Router      Age Seq#        CkSum  Route
10.0.0.0        2.2.2.2         214 0x80000001 0xd455 10.0.0.0/24
100.0.0.0       2.2.2.2         173 0x80000001 0x4886 100.0.0.0/24
30.0.0.0        3.3.3.3         208 0x80000001 0xb160 30.0.0.0/24
110.0.0.0       3.3.3.3         171 0x80000001 0xa719 110.0.0.0/24

                ASBR-Summary Link States (Area 0)

Link ID         ADV Router      Age Seq#        CkSum
4.4.4.4         3.3.3.3         171 0x80000001 0x72ac

                Router Link States (Area 2 [NSSA])

Link ID         ADV Router      Age Seq#        CkSum  Link count
3.3.3.3         3.3.3.3         175 0x80000004 0x686f 1
4.4.4.4         4.4.4.4         177 0x80000005 0xe46a 2

                Net Link States (Area 2 [NSSA])

Link ID         ADV Router      Age Seq#        CkSum
30.0.0.2        4.4.4.4         177 0x80000001 0xc827

                Summary Link States (Area 2 [NSSA])

Link ID         ADV Router      Age Seq#        CkSum  Route
10.0.0.0        3.3.3.3         172 0x80000001 0xde48 10.0.0.0/24
20.0.0.0        3.3.3.3         214 0x80000001 0x52cb 20.0.0.0/24
100.0.0.0       3.3.3.3         172 0x80000001 0x5279 100.0.0.0/24

                NSSA-external Link States (Area 2 [NSSA])
```

```
Link ID       ADV Router      Age Seq#       CkSum  Route
200.1.1.0     4.4.4.4        247 0x80000001 0x6cde N2 200.1.1.0/24 [0x0]

                AS External Link States

Link ID       ADV Router      Age Seq#       CkSum  Route
200.1.1.0     3.3.3.3        176 0x80000001 0x0156 E2 200.1.1.0/24  [0x0]
```

According to the OSPF database, the ABR in the NSSA area (Area 2) converts NSSA-external LSAs into AS External LSAs. Therefore, the other areas can normally learn external routes that are redistributed from the NSSA area (Area 2).

#Query the route table of Device2.

```
Device2#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

C   10.0.0.0/24 is directly connected, 00:02:53, gigabitethernet0
L   10.0.0.2/32 is directly connected, 00:02:53, gigabitethernet0
C   20.0.0.0/24 is directly connected, 00:02:51, gigabitethernet1
L   20.0.0.1/32 is directly connected, 00:02:51, gigabitethernet1
O   30.0.0.0/24 [110/2] via 20.0.0.2, 00:02:04, gigabitethernet1
O   100.0.0.0/24 [110/2] via 10.0.0.1, 00:02:04, gigabitethernet0
O   110.0.0.0/24 [110/3] via 20.0.0.2, 00:02:02, gigabitethernet1
C   127.0.0.0/8 is directly connected, 06:47:22, lo0
L   127.0.0.1/32 is directly connected, 06:47:22, lo0
OE  200.1.1.0/24 [150/20] via 20.0.0.2, 00:02:02, gigabitethernet1
```

Device2 has learnt the external routes that have been redistributed from the NSSA area (Area 2).

#Query the route table of Device1.

```
Device1#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

C   10.0.0.0/24 is directly connected, 00:02:29, gigabitethernet1
L   10.0.0.1/32 is directly connected, 00:02:29, gigabitethernet1
```

```
O   20.0.0.0/24 [110/2] via 10.0.0.2, 00:01:44, gigabitethernet1
O   30.0.0.0/24 [110/3] via 10.0.0.2, 00:01:41, gigabitethernet1
C   100.0.0.0/24 is directly connected, 01:53:00, gigabitethernet0
L   100.0.0.1/32 is directly connected, 01:53:00, gigabitethernet0
O   110.0.0.0/24 [110/4] via 10.0.0.2, 00:01:40, gigabitethernet1
C   127.0.0.0/8 is directly connected, 383:45:55, lo0
L   127.0.0.1/32 is directly connected, 383:45:55, lo0
```

According to the queried information, route 200.1.1.0/24 does not exist in the route table of Device1, indicating that the external route redistributed by Device4 has not been injected to the NSSA area (Area 1), while the routes of other areas have been added into the route table.

Step 4: Configure Device2, and introduce a default route to Area 1.

#Configure Device2. At this time, Device2 is the ABR of Area 1.

```
Device2(config)#router ospf 100
Device2(config-ospf)#router-id 2.2.2.2
Device2(config-ospf)#area 1 nssa default-information-originate
Device2(config-ospf)#exit
```

Note:

After the **area** *area-id* **nssa no-summary** command is executed on the ABR of the NSSA area, the area is called a totally NSSA area. At this time, the ABR generates a default route and flood the default route into the NSSA area. After the command is configured, the number of summary LSAs and corresponding inter-area routes will be further decreased. The devices in the area access a network outside the area or outside the AS through the default route.

Step 5: Check the result.

#Query OSPF LSDB of Device2.

```
Device2#show ip ospf database

            OSPF Router with ID (2.2.2.2) (Process ID 100)

                Router Link States (Area 0)

Link ID         ADV Router      Age Seq#       CkSum  Link count
2.2.2.2         2.2.2.2         455 0x80000004 0xe110 1
3.3.3.3         3.3.3.3         455 0x80000004 0xa345 1

                Net Link States (Area 0)

Link ID         ADV Router      Age Seq#       CkSum
```

```
20.0.0.2       3.3.3.3         461 0x80000001 0xf60d

               Summary Link States (Area 0)

Link ID        ADV Router      Age Seq#       CkSum  Route
10.0.0.0       2.2.2.2         492 0x80000001 0xd455 10.0.0.0/24
100.0.0.0      2.2.2.2         449 0x80000001 0x4886 100.0.0.0/24
30.0.0.0       3.3.3.3         487 0x80000001 0xb160 30.0.0.0/24
110.0.0.0      3.3.3.3         449 0x80000001 0xa719 110.0.0.0/24

               ASBR-Summary Link States (Area 0)

Link ID        ADV Router      Age Seq#       CkSum
4.4.4.4        3.3.3.3         449 0x80000001 0x72ac

               Router Link States (Area 1 [NSSA])

Link ID        ADV Router      Age Seq#       CkSum  Link count
1.1.1.1        1.1.1.1         456 0x80000005 0x8d0f 2
2.2.2.2        2.2.2.2         457 0x80000004 0x59ad 1

               Net Link States (Area 1 [NSSA])

Link ID        ADV Router      Age Seq#       CkSum
10.0.0.2       2.2.2.2         457 0x80000001 0x61ba

               Summary Link States (Area 1 [NSSA])

Link ID        ADV Router      Age Seq#       CkSum  Route
20.0.0.0       2.2.2.2         492 0x80000001 0x70b1 20.0.0.0/24
30.0.0.0       2.2.2.2         449 0x80000001 0xf71f 30.0.0.0/24
110.0.0.0      2.2.2.2         448 0x80000001 0xedd7 110.0.0.0/24

               NSSA-external Link States (Area 1 [NSSA])

Link ID        ADV Router      Age Seq#       CkSum  Route
0.0.0.0        2.2.2.2         31 0x80000001 0x5b42 N2 0.0.0.0/0 [0x0]
```

```
                AS External Link States

Link ID      ADV Router     Age Seq#      CkSum  Route
200.1.1.0      3.3.3.3        454 0x80000001 0x0156 E2 200.1.1.0/24  [0x0]
```

OSPF has generated a NSSA-external LSA for the default route.

#Query the route table of Device1.

```
Device1#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

OE  0.0.0.0/0 [150/1] via 10.0.0.2, 00:00:22, gigabitethernet1
C   10.0.0.0/24 is directly connected, 00:07:29, gigabitethernet1
L   10.0.0.1/32 is directly connected, 00:07:29, gigabitethernet1
O   20.0.0.0/24 [110/2] via 10.0.0.2, 00:06:44, gigabitethernet1
O   30.0.0.0/24 [110/3] via 10.0.0.2, 00:06:41, gigabitethernet1
C   100.0.0.0/24 is directly connected, 01:58:00, gigabitethernet0
L   100.0.0.1/32 is directly connected, 01:58:00, gigabitethernet0
O   110.0.0.0/24 [110/4] via 10.0.0.2, 00:06:40, gigabitethernet1
C   127.0.0.0/8 is directly connected, 383:50:55, lo0
L   127.0.0.1/32 is directly connected, 383:50:55, lo0
```

The route table of Device1 has learnt the default route 0.0.0.0, and Device1 communicates with the network outside the AS through the default route.

6.3.10. Configure OSPF to Coordinate with BFD

Network Requirements

- Configure OSPF for all devices.
- Enable the BFD detection function on the line between Device1 and Device3. If the line becomes faulty, BFD quickly detects the fault and notify OSPF of the fault. Then, OSPF switches the route to Device2 for communication.

Network Topology

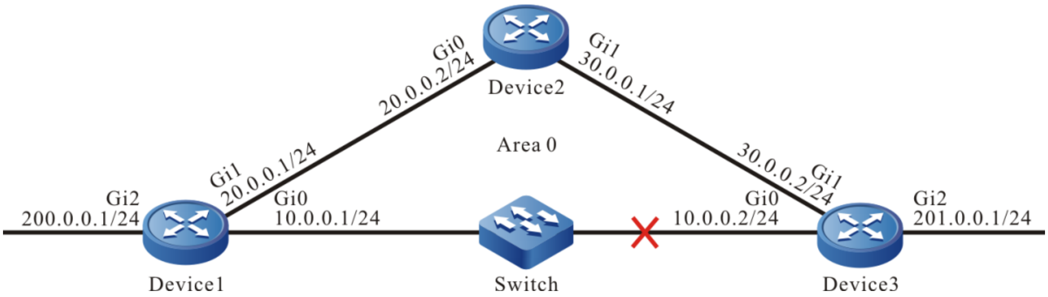

Figure 6-10 Networking for configuring OSPF to coordinate with BFD

Configuration Steps

Step 1: Configure the IP addresses of the interfaces. (Omitted)

Step 2: Configure an OSPF process.

#Configure Device1.

```
Device1#configure terminal
Device1(config)#router ospf 100
Device1(config-ospf)#router-id 1.1.1.1
Device1(config-ospf)#network 10.0.0.0 0.0.0.255 area 0
Device1(config-ospf)#network 20.0.0.0 0.0.0.255 area 0
Device1(config-ospf)#network 200.0.0.0 0.0.0.255 area 0
Device1(config-ospf)#exit
```

#Configure Device2.

```
Device2#configure terminal
Device2(config)#router ospf 100
Device2(config-ospf)#router-id 2.2.2.2
Device2(config-ospf)#network 20.0.0.0 0.0.0.255 area 0
Device2(config-ospf)#network 30.0.0.0 0.0.0.255 area 0
Device2(config-ospf)#exit
```

#Configure Device3.

```
Device3#configure terminal
Device3(config)#router ospf 100
Device3(config-ospf)#router-id 3.3.3.3
Device3(config-ospf)#network 10.0.0.0 0.0.0.255 area 0
Device3(config-ospf)#network 30.0.0.0 0.0.0.255 area 0
Device3(config-ospf)#network 201.0.0.0 0.0.0.255 area 0
Device3(config-ospf)#exit
```

Step 3: Configure OSPF to coordinate with BFD.

#Configure Device1.

```
Device1(config)#interface gigabitethernet0
Device1(config-if-gigabitethernet0)#ip ospf bfd
Device1(config-if-gigabitethernet0)#exit
```

#Configure Device3.

```
Device3(config)#interface gigabitethernet0
Device3(config-if-gigabitethernet0)#ip ospf bfd
Device3(config-if-gigabitethernet0)#exit
```

Step 4: Check the result.

#Query the OSPF neighbors and route table of Device1.

```
Device1#show ip ospf neighbor 3.3.3.3

OSPF process 100:
 Neighbor 3.3.3.3, interface address 10.0.0.2
    In the area 0 via interface gigabitethernet0, BFD enabled
    Neighbor priority is 1, State is Full, 5 state changes
    DR is 10.0.0.2, BDR is 10.0.0.1
    Options is 0x42 (-|O|-|-|-|-|E|-)
    Dead timer due in 00:00:31
    Neighbor is up for 00:02:46
    Database Summary List 0
    Link State Request List 0
    Link State Retransmission List 0
    Crypt Sequence Number is 0

    Thread Inactivity Timer on
    Thread Database Description Retransmission off, 0 times
    Thread Link State Request Retransmission off, 0 times
    Thread Link State Update Retransmission off, 0 times
    Hello received 17 sent 19, DD received 3 sent 3
    LS-Req received 0 sent 0, LS-Upd received 1(LSA: 2) sent 1(LSA: 1)
    LS-Ack received 1 sent 0, Discarded 0

Device1#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

C   10.0.0.0/24 is directly connected, 00:01:09, gigabitethernet0
L   10.0.0.1/32 is directly connected, 00:01:09, gigabitethernet0
C   20.0.0.0/24 is directly connected, 00:55:37, gigabitethernet1
L   20.0.0.1/32 is directly connected, 00:55:37, gigabitethernet1
O   30.0.0.0/24 [110/2] via 20.0.0.2, 00:02:50, gigabitethernet1
                [110/2] via 10.0.0.2, 00:01:30, gigabitethernet0
```

```
C   127.0.0.0/8 is directly connected, 05:51:09, lo0
L   127.0.0.1/32 is directly connected, 05:51:09, lo0
C   200.0.0.0/24 is directly connected, 00:55:12, gigabitethernet2
L   200.0.0.1/32 is directly connected, 00:55:12, gigabitethernet2
O   201.0.0.0/24 [110/2] via 10.0.0.2, 00:01:30, gigabitethernet0
```

According to the OSPF neighbor information, BFD has been enabled, and route 201.0.0.0/24 selects the line between Device1 and Device3 with priority for communication.

#Query the BFD session of Device1.

```
Device1#show bfd session detail
Total session number: 1
OurAddr            NeighAddr            LD/RD           State       Holddown
interface
10.0.0.1        10.0.0.2          7/14        UP        5000        gigabitethernet0
Type:direct
Local State:UP  Remote State:UP  Up for: 0h:2m:37s  Number of times UP:1
Send Interval:1000ms  Detection time:5000ms(1000ms*5)
Local Diag:0  Demand mode:0  Poll bit:0
MinTxInt:1000  MinRxInt:1000  Multiplier:5
Remote MinTxInt:1000  Remote MinRxInt:1000  Remote Multiplier:5
Registered protocols:OSPF
```

According to the queried information, OSPF has been configured successfully to coordinate with BFD, and the session has been normally set up.

#If the line between Device1 and Device3 becomes faulty, BFD quickly detects the fault and informs OSPF of the fault. OSPF then switch the route to Device2 for communication. Query the route table of Device1.

```
%BFD-5-Session [10.0.0.2,10.0.0.1,gigabitethernet0,10] DOWN (Detection time expired)
%OSPF-5-ADJCHG: Process 100 Nbr [gigabitethernet0:10.0.0.1-3.3.3.3] from Full to
Down,KillNbr: BFD session down


Device1#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

C   10.0.0.0/24 is directly connected, 00:01:59, gigabitethernet0
L   10.0.0.1/32 is directly connected, 00:01:59, gigabitethernet0
C   20.0.0.0/24 is directly connected, 00:56:13, gigabitethernet1
L   20.0.0.1/32 is directly connected, 00:56:13, gigabitethernet1
O   30.0.0.0/24 [110/2] via 20.0.0.2, 00:03:40, gigabitethernet1
```

```
C   127.0.0.0/8 is directly connected, 05:52:41, lo0
L   127.0.0.1/32 is directly connected, 05:52:41, lo0
C   200.0.0.0/24 is directly connected, 00:56:02, gigabitethernet2
L   200.0.0.1/32 is directly connected, 00:56:02, gigabitethernet2
O   201.0.0.0/24 [110/3] via 20.0.0.2, 00:00:06, gigabitethernet1
```

The action of Device3 is similar to that of Device1.

6.3.11. Configure OSPF Fast Re-routing

Network Requirements

- All devices configure the OSPF protocol.
- Device1 learns the OSPF route 192.168.1.1/32 from Device2 and Device3 at the same time. Device1 first uses the line with Device3 to forward the packet. Similarly, Device3 learns the OSPF route 100.1.1.1/32 from Device1 and Device2 at the same time. Device3 first uses the line with Device1 to forward the packet.
- Device1 and Device3 enable the OSPF fast re-routing. After the line between Device1 and Device3 fails, the service can switch to Device2 for communication fast.

Network Topology

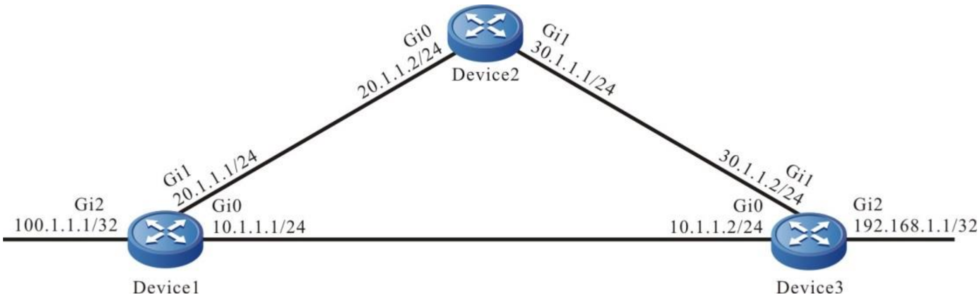

Figure 6-11 Configure the OSPF fast re-routing

Configuration Steps

Step 1: Configure the IP addresses of the interfaces. (Omitted)

Step 2: Configure OSPF.

#Configure Device1. Configure the OSPF process and make the interface cover area 0.

```
Device1#configure terminal
Device1(config)#router ospf 100
Device1(config-ospf)#router-id 1.1.1.1
Device1(config-ospf)#network 10.1.1.0 0.0.0.255 area 0
Device1(config-ospf)#network 20.1.1.0 0.0.0.255 area 0
Device1(config-ospf)#network 100.1.1.1 0.0.0.0 area 0
Device1(config-ospf)#exit
```

#Configure Device2. Configure the OSPF process and make the interface cover area 0.

```
Device2#configure terminal
Device2(config)#router ospf 100
```

```
Device2(config-ospf)#router-id 2.2.2.2
Device2(config-ospf)#network 20.1.1.0 0.0.0.255 area 0
Device2(config-ospf)#network 30.1.1.0 0.0.0.255 area 0
Device2(config-ospf)#exit
```

#Configure Device3. Configure the OSPF process and make the interface cover area 0.

```
Device3#configure terminal
Device3(config)#router ospf 100
Device3(config-ospf)#router-id 3.3.3.3
Device3(config-ospf)#network 10.1.1.0 0.0.0.255 area 0
Device3(config-ospf)#network 30.1.1.0 0.0.0.255 area 0
Device3(config-ospf)#network 192.168.1.1 0.0.0.0 area 0
Device3(config-ospf)#exit
```

Step 3: Configure the routing policy.

#Configure Device1: configure route-map to call the ACL only matching 192.168.1.1/32, while the other network is filtered. The route matching the match rule applies the backup next-hop interface gigabitethernet1 and the next-hop address is 20.1.1.2.

```
Device1(config)#ip access-list standard 1
Device1(config-std-nacl)#permit 192.168.1.1 0.0.0.0
Device1(config-std-nacl)#exit
Device1(config)#route-map ipfrr_ospf
Device1(config-route-map)#match ip address 1
Device1(config-route-map)#set fast-reroute backup-interface gigabitethernet1 backup-nexthop 20.1.1.2
Device1(config-route-map)#exit
```

#Configure Device3: configure route-map to call the ACL only matching 100.1.1.1/32, while the other network is filtered. The route matching the match rule applies the backup next-hop interface gigabitethernet1 and the next-hop address is 30.1.1.1.

```
Device3(config)#ip access-list standard 1
Device3(config-std-nacl)#permit 100.1.1.1 0.0.0.0
Device3(config-std-nacl)#exit
Device3(config)#route-map ipfrr_ospf
Device3(config-route-map)#match ip address 1
Device3(config-route-map)#set fast-reroute backup-interface gigabitethernet1 backup-nexthop 30.1.1.1
Device3(config-route-map)#exit
```

Step 4: Configure the fast re-routing.

#Configure Device1 to enable the OSPF fast re-routing.

```
Device1(config)#router ospf 100
Device1(config-ospf)#ipfrr route-map ipfrr_ospf
Device1(config-ospf)#exit
```

#Configure Device3 to enable the OSPF fast re-routing

```
Device3(config)#router ospf 100
Device3(config-ospf)#ipfrr route-map ipfrr_ospf
Device3(config-ospf)#exit
```

Step 5: Check the result.

#View the route table of Device1.

```
Device1#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
    U - Per-user Static route
    O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

C   10.1.1.0/24 is directly connected, 00:22:44, gigabitethernet0
L   10.1.1.1/32 is directly connected, 00:22:44, gigabitethernet0
C   20.1.1.0/24 is directly connected, 06:39:56, gigabitethernet1
L   20.1.1.1/32 is directly connected, 06:39:56, gigabitethernet1
O   30.1.1.0/24 [110/2] via 20.1.1.2, 00:01:51, gigabitethernet1
              [110/2] via 10.1.1.2, 00:00:16, gigabitethernet0
C   127.0.0.0/8 is directly connected, 31:14:38, lo0
L   127.0.0.1/32 is directly connected, 31:14:38, lo0
LC   100.1.1.1/32 is directly connected, 03:14:47, loopback0
O   192.168.1.1/32 [110/2] via 10.1.1.2, 00:00:04, gigabitethernet0
```

#View the fast re-route table of Device1 and you can see that there is the route of the network 192.168.1.1/32 and the next-hop interface is gigabitethernet1.

```
Device1#show ip frr route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
    U - Per-user Static route
    O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

O   192.168.1.1/32 [110/0] via 20.1.1.2, 00:00:08, gigabitethernet1
```

#View the backup next-hop information of Device1 and the fast re-routing backup interface is gigabitethernet1.

```
Device1#show nexthop frr detail
Index                : 225
```

```
Type                : FRR
Reference Count         : 1
Active Path           : master
Nexthop Address         : 10.1.1.2
Interface             : gigabitethernet0
Interface Vrf           : global
Channel ID            : 10
Link Header Length      : 18
Link Header           : 00017a123453201201010101810000010800
Action                : FORWORDING
Slot                  : 0
BK Nexthop Address      : 20.1.1.2
BK Interface            : gigabitethernet1
BK Interface Vrf        : global
BK Channel ID           : 11
BK Link Header Length   : 18
BK Link Header          : 00017a455449201201010102810000020800
BK Action               : FORWORDING
BK Slot                 : 0

Total 1 entries.
```

#After the line between Device1 and Device3 fails, the system can fast detect and switch to Device2 for communication. View the route table and fast re-route table of Device1. The egress interface to the destination network 192.168.1.1/32 in the route table is switched to the backup interface gigabitethernet1.

```
Device1#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external


C   10.1.1.0/24 is directly connected, 00:22:44, gigabitethernet0
L   10.1.1.1/32 is directly connected, 00:22:44, gigabitethernet0
C   20.1.1.0/24 is directly connected, 06:39:56, gigabitethernet1
L   20.1.1.1/32 is directly connected, 06:39:56, gigabitethernet1
O   30.1.1.0/24 [110/2] via 20.1.1.2, 00:01:51, gigabitethernet1
               [110/2] via 10.1.1.2, 00:00:16, gigabitethernet0
C   127.0.0.0/8 is directly connected, 31:14:38, lo0
L   127.0.0.1/32 is directly connected, 31:14:38, lo0
```

```
LC   100.1.1.1/32 is directly connected, 03:14:47, loopback0
O   192.168.1.1/32 [110/3] via 20.1.1.2, 00:00:04, gigabitethernet1
```

The processing mode of Device3 is similar with Device1.

7. OSPFV3

7.1. Overview

OSPFv3 is the abbreviation of OSPF (Open Shortest Path First) Version 3. It mainly provides support for IPv6, follows RFC2328, RFC2740, and supports OSPF extensions defined by other related RFCs.

OSPFv3 is basically the same as OSPFv2, but there are some corresponding modifications for different IP protocols and address families. Their differences are mainly manifested in:

- OSPFv3 runs based on the link, and OSPFv2 runs based on the segment
- Each OSPFv3 link supports multiple instances
- OSPFv3 identifies the neighbor via Router ID, and OSPFv2 identifies the neighbor via the IP address.

7.2. OSPFv3 Function Configuration

Table 7-1 OSPFv3 function list

Configuration Tasks	
Configure basic OSPFv3 functions.	Enable OSPFv3.
Configure OSPFv3 areas.	Configure an OSPFv3 NSSA area.
	Configure an OSPFv3 Stub area.
	Configure an OSPFv3 virtual link.
Configure the OSPFv3 network type.	Configure the network type of an OSPFv3 interface to broadcast.
	Configure the network type of an OSPFv3 interface to P2P.
	Configure the network type of an OSPFv3 interface to NBMA.
	Configure the network type of an OSPFv3 interface to P2MP.

Configuration Tasks	
Configure the OSPFv3 network authentication.	Configure OSPFv3 area authentication.
	Configure OSPFv3 interface authentication.
Configure OSPFv3 route generation.	Configure OSPFv3 to redistribute routes.
	Configure the default OSPFv3 route.
Configure OSPFv3 route control.	Configure route summary on inter-area OSPFv3 routes.
	Configure OSPFv3 external route summary.
	Configure route filtering on inter-area OSPFv3 routes.
	Configure OSPFv3 external route filtration.
	Configure OSPFv3 route installation filtration.
	Configure the cost value of an OSPFv3 interface.
	Configure the OSPFv3 reference bandwidth.
	Configure the OSPFv3 administrative distance.
	Configure the maximum number of OSPFv3 load balancing routes.

Configuration Tasks	
Configure OSPFv3 network optimization.	Configure the keep-alive time of an OSPFv3 neighbor.
	Configure an OSPFv3 passive interface.
	Configure an OSPFv3 demand circuit.
	Configure the priority of an OSPFv3 interface.
	Configure the MTU of an OSPFv3 interface.
	Configure the LSA transmit delay of an OSPFv3 interface.
	Configure OSPFv3 LSA retransmission.
	Configure OSPFv3 SPF calculation time.
Configure OSPFv3 fast re-routing	Configure OSPFv3 fast re-routing
Configure OSPFv3 to coordinate with BFD.	Configure OSPFv3 to coordinate with BFD.

7.2.1. Configure Basic OSPFv3 Functions

Before configuring OSPFv3 functions, you must first enable the OSPFv3 protocol before the other functions can take effect.

Configuration Conditions

Before configuring the basic OSPFv3 functions, ensure that:

- The link layer protocol has been configured to ensure normal communication at the link layer.
- Enable the IPv6 forwarding functions

Enable OSPFv3

To enable OSPFv3, you must create an OSPFv3 process, specify the Router ID of the process, and enable the OSPFv3 protocol on the interface.

A device that runs the OSPFv3 protocol must have a Router ID, which is used to uniquely identify a device in an OSPFv3 AS. You must ensure that the Router ID is unique in an AS; otherwise, setup of neighbors and route learning are affected. In OSPFv3, you need to manually configure one Router ID of the IPv4 address format.

OSPFv3 supports multiple processes, and uses the process number to identify one process. Different processes are independent of each other and have no influence on each other.

Table 7-2 Enable the OSPFv3 protocol

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Create an OSPFv3 process and enter the OSPF configuration mode.	**ipv6 router ospf** *process-id* [**vrf** *vrf-name*]	Mandatory. Enable the OSPFv3 process or enable the OSPF process from VRF. By default, the OSPFv3 protocol is disabled. If you enable OSPFv3 from VRF, the OSPFv3 process that belongs to a VRF can manage only interfaces under the VRF.
Configure the Router ID of the OSPFv3 process.	**router-id** *ipv4-address*	Mandatory
Return to the global configuration mode	**exit**	-
Enter the interface configuration mode	**interface** *interface-name*	-
Configure enabling the OSPFv3 protocol on the interface	**ipv6 router ospf** *process-id* **area** *area-id* [**instance-id** *instance-id*]	Mandatory By default, the OPSFv3 protocol is disabled on the interface.

7.2.2. Configure the OSPFv3 Area

To prevent a large amount of database information from occupying too much CPU and memory, you can divide an OSPFv3 AS into multiple areas. An area can be identified with a 32-bit area ID, a decimal number in the range of 0-4294967295, or an IP address in the range of 0.0.0.0-255.255.255.255. Area 0 or 0.0.0.0 represents an OSPFv3 backbone area, while other non-zero areas are non-backbone areas. All routing information between areas must be forwarded through the backbone area. Non-backbone areas cannot directly exchange routing information.

OSPFv3 defines several types of routers:

- Internal router: All interfaces belong to the devices in one area.
- Area Border Router (ABR): It is connected to devices from different areas.
- Autonomous System Boundary Router (ASBR): It is a device that introduces external routes to the OSPFv3 AS.

Configuration Condition

Before configuring an OSPFv3 area, ensure that:

- Enable the IPv6 forwarding function
- Enable the OSPFv3 protocol

Configure an OSPFv3 NSSA Area

A Not-So-Stub-Area (NSSA) does not allow injection of Type-5 Link State Advertisement (LSA) but it allows injection of Type-7 LSA. External routes can be introduced to an NSSA area through redistribution of configuration. The ASBR in the NSSA area generate Type-7 LSAs and flood LSAs to the NSSA area. The ABR in an NSSA area converts Type-7 LSAs into Type-5 LSAs, and floods the converted Type-5 LSAs into the entire AS.

The OSPFv3 NSSA area that is configured by using the **area** *area-id* **nssa no-summary** command is called a totally NSSA area. An OSPFv3 totally NSSA area does not allow cross-area routes to flood in the area. At this time, the ABR generates a default route and flood it into the NSSA area. The devices in the NSSA area access a network outside the area through the default route.

Table 7-3 Configure an OSPFv3 NSSA area

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the OSPFv3 configuration mode.	**ipv6 router ospf** *process-id* [**vrf** *vrf-name*]	-
Configure an NSSA area.	**area** *area-id* **nssa** [**no-redistribution** / **no-summary** / **default-information-originate** [**metric** *metric-value* / **metric-type** *type-value*]]	Mandatory By default, an OSPFv3 area is not the NSSA area.

Note:

- A backbone area cannot be configured as an NSSA area.
- All devices in one NSSA area must be configured as NSSA areas, because devices with different area types cannot form neighbor relations.

Configure an OSPFv3 Stub Area

A Stub area does not allow external route outside an AS to flood in the area so as to reduce the size of the link status database. After an area is configured as a Stub area, the ABR which is located at the Stub border generates a default route and flood the route into the Stub area. The devices in the Stub area access a network outside the area through the default route.

The OSPFv3 Stub area that is configured by using the **area** *area-id* **stub no-summary** command is called a totally Stub area. An OSPFv3 totally Stub area does not allow inter-area routes and external routes to flood in the area. The devices in the area access a network outside the area and outside the OSPFv3 AS through the default route.

Table 7-4 Configure an OSPFv3 Stub area

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the OSPFv3 configuration mode.	**ipv6 router ospf** *process-id* [**vrf** *vrf-name*]	-
Configure a Stub area.	**area** *area-id* **stub** [**no-summary**]	Mandatory By default, an OSPFv3 area is not the Stub area.

Note:

- A backbone area cannot be configured as a Stub area.
- All devices in one Stub area must be configured as Stub areas, because devices with different area types cannot form neighbor relations.

Configure an OSPFv3 Virtual Link

The non-backbone areas in OSPFv3 must synchronize and exchange data through the backbone area. Therefore, all non-backbone areas must keep connected with the backbone area.

If the requirement fails to be meet in certain cases, you can solve the problem by configuring a virtual link. After configuring a virtual link, you can configure an authentication mode for the virtual link and modify the Hello interval. The meanings of the parameters are the same as the meanings of the parameter of common OSPFv3 interfaces.

Table 7-5 Configure an OSPFv3 virtual link

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-

Step	Command	Description
Enter the OSPFv3 configuration mode.	**ipv6 router ospf** *process-id* [**vrf** *vrf-name*]	-
Configure an OSPFv3 virtual link.	**area** *transit-area-id* **virtual-link** *neighbor-id* [**dead-interval** *seconds* / **hello-interval** *seconds* / **retransmit-interval** *seconds* / **transmit-delay** *seconds*]	andatory. By default, no virtual link is created.

Note:

- A virtual link must be configured between two ABRs.
- Two ABRs on which the virtual link is configured must be in the same public area. This area is also called the transit area of the virtual link.
- The transit area of a virtual link must not be a Stub area or NSSA area.

7.2.3. Configure OSPFv3 Network Type

According to the link protocol types, OSPFv3 classifies networks into four types:

- Broadcast Network: When the link protocol of the network is Ethernet or Fiber Distributed Data Interface (FDDI), the default OSPFv3 network type is broadcast.
- Point To Point Network (P2P Network): When the link protocol is Point to Point Protocol (PPP), Link Access Procedure Balanced (LAPB), or High-level Data Link Control (HDLC), the default OSPFv3 network type is P2P.
- Non-Broadcast Multi-Access Network (NBMA Network): When the link protocol is ATM, frame relay, or X.25, the default OSPFv3 network type is NBMA.
- Point To Multi-Point Network (P2MP): No link protocol will be regarded by OSPFv3 as the P2MP network by default. Usually, the NBMA network that is not totally connected is configured as the OSPFv3 P2MP network.

You can modify the network type of an OSPFv3 interface according to the actual requirement. The network types of the interfaces through which OSPFv3 neighbors are set up must be the same; otherwise, normal learning of routes is affected.

Configuration Condition

Before configuring the OSPFv3 network type, ensure that:

- Enable the IPv6 forwarding function
- Enable the OSPFv3 protocol

Configure the Network Type of an OSPFv3 Interface to Broadcast

A broadcast network supports multiple devices (more than two devices). These devices can exchange information with all the devices in the network. OSPFv3 uses Hello packets to dynamically discover neighbors.

Table 7-6 Configure the network type of an OSPFv3 interface to broadcast

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the interface configuration mode.	**interface** *interface-name*	-
Configure the network type of an OSPFv3 interface to broadcast.	**ipv6 ospf network broadcast**	Mandatory. By default, the network type of an OSPFv3 interface is determined by the link layer protocol.

Configure the Network Type of an OSPFv3 Interface to P2P

A P2P network is a network that consists of two devices. Each device is located at one end of a P2P link. OSPFv3 uses Hello packets to dynamically discover neighbors.

Table 7-7 Configure the network type of an OSPFv3 interface to P2P

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the interface configuration mode.	**interface** *interface-name*	-
Configure the OSPFv3 network type to P2P.	**ipv6 ospf network point-to-point**	Mandatory. By default, the network type of an OSPFv3 interface is determined by the link layer protocol.

Configure the Network Type of an OSPFv3 Interface to NBMA

An NBMA network supports multiple devices (more than two devices), but the devices does not have the broadcast capability, therefore, you must specify a neighbor manually.

Table 7-8 Configure the network type of an OSPFv3 interface to NBMA

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the interface configuration mode.	**interface** *interface-name*	-
Configure the network type of the OSPFv3 interface to NBMA.	**ipv6 ospf network non-broadcast**	Mandatory. By default, the network type of an OSPFv3 interface is determined by the link layer protocol.
Configure a neighbor for the NBMA network.	**ipv6 ospf neighbor** *neighbor-ipv6-address* [**priority** *priority-value* / **poll-interval** *interval-value* / **cost** *cost-value*] [**instance-id** *instance-id*]	Mandatory. In an NBMA network, a neighbor must be specified manually.

Configure the Network Type of an OSPFv3 Interface to P2MP

When an NBMA network is not fully connected, you can configure its network type to P2MP to save network overhead. If the network type is configured to P2MP unicast, you need to specify a neighbor manually.

Table 7-9 Configure the network type of an OSPFv3 interface to P2MP

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the interface configuration mode.	**interface** *interface-name*	-
Configure the OSPFv3 network type to P2MP.	**ipv6 ospf network point-to-multipoint** [**non-broadcast**]	andatory. By default, the network type of an OSPFv3 interface is determined by the link layer protocol.

Step	Command	Description
Configure a neighbor for the P2MP unicast network.	**ipv6 ospf neighbor** *neighbor-ipv6-address* [**priority** *priority-value* / **poll-interval** *interval-value* / **cost** *cost-value*] [**instance-id** *instance-id*]	If the interface network type is set to P2MP unicast, it is mandatory.

7.2.4. Configure OSPFv3 Network Authentication

To prevent information leakage or malicious attacks to OSPFv3 devices, all packet interaction between OSPFv3 neighbors has the authentication capability. The encrypted authentication types and algorithms include: NULL (no authentication), SHA1 authentication, and MD5 authentication, which is specified by the IPSec encrypted authentication policy.

After configuring authentication, IPSec security features encrypt and authenticate OSPFv3 protocol packets.The OSPFv3 protocol can receive packets only after decryption authentication. Therefore, the OSPFv3 interfaces which establish the adjacency relationship must have the same authentication method, Spi ID, and IPSec encryption authentication policy of authentication password configuration. The OSPFv3 authentication mode can be configured on the area and interface, and its priority is from low to high: area authentication, interface authentication. That is, first use the interface authentication mode, and then, use the area authentication mode.

Configuration Condition

Before configuring OSPFv3 network authentication, ensure that:

- The IPv6 forwarding function is enabled.
- The OSPFv3 protocol is enabled.

Configure OSPFv3 Area Authentication

Configuring the area authentication in the OSPFv3 process area can make all interfaces in the area use the area authenticaton mode, and effectively avoid configuring the same network authentication mode in the interface repeatedly.

Table 7-10 Configure OSPFv3 area authentication

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the OSPFv3 configuration mode.	**ipv6 router ospf** *process-id* [**vrf** *vrf-name*]	-

Step	Command	Description
Configure the area authentication mode.	**area** *area-id* **ipsec-tunnel** *tunnel-name*	Mandatory By default, OSPFv3 is not configured with the area authentication.

Configure OSPFv3 Interface Authentication

If an interface has multiple OSPFv3 instances, you can specify the authentication mode and password for one instance. If you do not specify the interface authentication instance in the interface, adopt the specified authentication mode in the area.

Table 7-11 Configure OSPFv3 interface authentication

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the interface configuration mode.	**interface** *interface-name*	-
Configure the interface authentication mode.	**ipv6 ospf ipsec-tunnel** *tunnel-name*{instance-id *instance-id*}	Mandatory By default, OSPFv3 is not configured with the interface authentication mode.

7.2.5. Configure OSPFv3 Route Generation

Configuration Condition

Before configuring OSPFv3 route generation, ensure that:

- The IPv6 forwarding function is enabled.
- The OSPFv3 protocol is enabled.

Configure OSPFv3 Route Re-distribution

If multiple routing protocols run on one device, routes of other protocols can be introduced to OSPF through redistribution. By default, class-2 external routes of OSPFv3 are generated with the route metric 20. When you introduces external routes through redistribution, you can modify the external route type, metric, and tag field, and associate the specified routing policy to perform route control and management.

Table 7-12 Configure OSPFv3 route re-distribution

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the OSPFv3 configuration mode.	**ipv6 router ospf** *process-id* [**vrf** *vrf-name*]	-
Configure OSPFv3 route re-distribution	**redistribute** *routing-protocol* [*protocol-id-or-name*] [**metric** *metric-value* / **metric-type** *type-value* / **tag** *tag-value* / **route-map** *map-name* / **match** *route-type*]	Mandatory By default, OSPFv3 is not configured with the route re-distribution.
Configure the metric of the OSPFv3 external route.	**default-metric** *metric-value*	Optional

Note:

- If the metric value of external routes are configured by using both the **redistribute** *protocol* [*protocol-id*] **metric** command and the **default-metric** command, the value that is configured by using the former command has a higher priority.

Configure OSPFv3 Default Route

After an OSPFv3 Stub area or a totally NSSA areas is configured, a Type-3 default route is generated. For an NSSA area, no default route is automatically generated. You can use the **area** *area-id* **nssa default-information-originate** command to introduce a Type-7 default route to the NSSA area.

OSPFv3 cannot use the **redistribute** command to introduce a Type-5 default route. To do this, use the **default-information originate** [**always**] command.

Table 7-13 Configure the default OSPFv3 route

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the OSPFv3 configuration mode.	**ipv6 router ospf** *process-id* [**vrf** *vrf-name*]	-

Step	Command	Description
Configure OSPFv3 to introduce a default route.	**default-information originate** [**always** / **metric** *metric-value* / **metric-type** *metric-type* / **route-map** *route-map-name*]	Mandatory. By default, no external default route is introduced to an OSPFv3 AS. The default metric of the introduced default route is 1, and the type is external type 2. The field **always** means to force the OSPFv3 AS to generate a default route; otherwise, the default route is generated only when there is a default route in the local routing table.

7.2.6. Configure OSPFv3 Route Control

Configuration Condition

Before configuring OSPFv3 route control, ensure that:

- The IPv6 forwarding function is enabled.
- The OSPFv3 protocol is enabled.

Configure Route Summary between OSPFv3 Areas

When an ABR in OSPFv3 advertises inter-area routes to other areas, it advertises each route separately in the form of Type-3 LSA. You can use the inter-area route summary function to summarize some continuous network segments to form a summary route. Then the ABR advertises the summary route, reducing the size of OSPFv3 databases.

Table 7-14 Configure route summary between OSPFv3 areas

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the OSPFv3 configuration mode.	**ipv6 router ospf** *process-id* [**vrf** *vrf-name*]	-

Step	Command	Description
Configure the route summary between OSPFv3 areas	**area** *area-id* **range** ipv6-*prefix*/*prefix-length* [**advertise** | **not-advertise**]	Mandatory By default, ABR does not perform the route summary between the areas.

Note:

- The route summary function between OSPFv3 areas is valid only for ABRs.
- By default, the minimum cost value among the cost values of the routes is used as the cost value of the route summary.

Configure OSPFv3 External Route Summary

When OSPF redistributes external routes, it advertises each route separately in the form of external LSA. You can use the external route summary function to summarize some continuous network segments to form a summary route. Then OSPF advertises the summary route, reducing the size of OSPF databases.

If you run the **summary-address** command on an ASBR, you can summarize all Type-5 LSAs and Type-7 LSAs within the address range.

Table 7-15 Configure OSPFv3 external route summary

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the OSPFv3 configuration mode.	**ipv6 router ospf** *process-id* [**vrf** *vrf-name*]	-
Configure OSPFv3 to summarize external routes.	**summary-prefix** *ipv6-prefix*/*prefix-length* [**not-advertise** | **tag** *tag-value*]	Mandatory. By default, an ABR does not summarize external routes.

Note:

- The OSPFv3 external route summary function is valid only for ASBRs.

Configure Route Filtering between OSPFv3 Areas

When an ABR receives inter-area routes, it performs filtration in the incoming direction based on an ACL or prefix list. When the ABR advertises inter-area routes, it performs filtration in the outgoing direction based on an ACL or prefix list.

Table 7-16 Configure route filteringbetween OSPFv3 areas

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the OSPFv3 configuration mode.	**ipv6 router ospf** *process-id* [**vrf** *vrf-name*]	-
Configure the route filtering between OSPFv3 areas	**area** *area-id* **filter-list** { **access** { *access-list-name* | *access-list-number }* | **prefix** *prefix-list-name* } { **in** | **out** }	Mandatory By default, ABR does not perform the route filtering between areas.

Note:

- The OSPFv3 inter-area route filtering function is valid only for ABRs.

Configure OSPFv3 External Route Filtration

Configuring OSPFv3 external route filtering is to apply an ACL or prefix list to allow or not allow external routes of an OSPFv3 AS to flood into the OSPF AS.

Table 7-17 Configure OSPFv3 external route filtration

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the OSPFv3 configuration mode.	**ipv6 router ospf** *process-id* [**vrf** *vrf-name*]	-
Configure the OSPFv3 external route filtering	**distribute-list** { **access** { *access-list-name* | *access-list-number }* | **prefix** *prefix-list-name* } **out** [*routing-protocol* [*process-id*]]	Mandatory By default, ASBR does not perform the external route filtering.

Note:

- The OSPFv3 external route filtering function is valid only for ASBRs.

Configure OSPFv3 Route Installation Filtration

After OSPFv3 calculates routes through LSA, to prevent certain routes from being added into the routing table, OSPFv3 filters the calculated OSPFv3 route information.

Table 7-18 Configure OSPFv3 route installation filtration

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the OSPFv3 configuration mode.	**ipv6 router ospf** *process-id* [**vrf** *vrf-name*]	-
Configure OSPFv3 route installation filtering	**distribute-list** { **access** { *access-list-name* **|** *access-list-number }* | **gateway** *prefix-list-name1* **| prefix** *prefix-list-name2* [**gateway** *prefix-list-name3*] | **route-map** *route-map-name*} **in [** *interface-name* **]**	Mandatory By default, do not configure OSPFv3 route installation filtering.

Note:

- Filtration based on prefix, gateway, and route-map is mutually exclusive with filtration based on ACL. For example, if you have configured filtration based on prefix, you cannot configure filtration based on ACL again.
- Filtration based on route-map and prefix is mutual exclusive with filtration based on gateway.
- Filtration based on prefix and filtration based on gateway overwrite each other.

Configure the Cost Value of an OSPFv3 Interface

By default, the cost of an OSPFv3 interface is calculated based on the following formula: Reference bandwidth/Interface bandwidth.

Table 7-19 Configure the cost value of an OSPFv3 interface

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the interface configuration mode.	**interface** *interface-name*	-

Step	Command	Description
Configure the cost value of an OSPFv3 interface.	**ipv6 ospf cost** *cost* [**instance-id** *instance-id*]	Optional. By default, the cost value is calculated through the formula Reference bandwidth/Interface bandwidth.

Configure the OSPFv3 Reference Bandwidth

The reference bandwidth of an interface is used to calculate the cost value of the interface. The default value is 100Mbit/s. The formula for calculating the cost value of the OSPFv3 interface is: Reference bandwidth/Interface bandwidth. If the calculation result is larger than 1, use the integer part. If the calculation result is smaller than 1, use the value 1. Therefore, in a network whose bandwidth is larger than 100Mbit/s, the optimal route fails to be selected. In this case, you can use the **auto-cost reference-bandwidth** command to configure a proper reference bandwidth to solve the problem.

Table 7-20 Configure the OSPFv3 reference bandwidth

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the OSPFv3 configuration mode.	**ipv6 router ospf** *process-id* [**vrf** *vrf-name*]	-
Configuring the OSPFv3 reference bandwidth.	**auto-cost reference-bandwidth** *reference-bandwidth*	Optional. By default, the reference bandwidth is 100Mbit/s

Configure the OSPFv3 Administrative Distance

An administrative distance is used to indicate the reliability of the routing protocol. If the routes to the same destination network are learnt by different routing protocols, the route with the smallest administrative distance is selected first.

Table 7-21 Configure the OSPFv3 administrative distance

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-

Step	Command	Description
Enter the OSPFv3 configuration mode.	**ipv6 router ospf** *process-id* [**vrf** *vrf-name*]	-
Configure the OSPFv3 administrative distance.	**distance [ospf** { **external** *distance* / **inter-area** *distance* / **intra-area** *distance* } | *distance*]	Optional. By default, the administrative distance of intra-area and inter-area OSPFv3 routes is 110, and the administrative distance of external routes is 150

Configure the Maximum Number of OSPFv3 Load Balancing Routes

If multiple equivalent paths are available to reach the same destination, load balancing is achieved. This improves the utility rate of links and reduces the load of the links.

Table 7-22 Configure the maximum number of OSPFv3 load balancing routes

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the OSPFv3 configuration mode.	**ipv6 router ospf** *process-id* [**vrf** *vrf-name*]	-
Configure the maximum number of OSPFv3 load balancing routes.	**maximum-paths** *max-number*	Optional. By default, the maximum number of OSPFv3 load balancing routes is 4.

7.2.7. Configure OSPFv3 Network Optimization

Configuration Condition

Before configuring OSPFv3 network optimization, ensure that:

- The IPv6 forwarding function is enabled.
- The OSPFv3 protocol is enabled.

Configure the Keep-alive Time of an OSPFv3 Neighbor

OSPFv3 Hello packets are used to set up neighbor relations and keep the relations alive. The default transmission interval of Hello packets is determined by the network type. For broadcast networks and P2P networks, the default transmission interval of Hello packets is 10s. For P2MP networks and NBMA networks, the default transmission interval of Hello packets is 30s.

Neighbor dead time is used to determine the validity of a neighbor. By default, the neighbor dead time is four times the Hello interval. If an OSPFv3 device fails to receive Hello packets from a neighbor after the neighbor dead time times out, the OSPFv3 device regards the neighbor as invalid, and then it deletes the neighbor in an active manner.

Table 7-23 Configure the keep-alive time of an OSPFv3 neighbor

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the interface configuration mode.	**interface** *interface-name*	-
Configure an OSPFv3 Hello interval.	**ipv6 ospf hello-interval** *interval-value* [**instance-id** *instance-id*]	Optional. The default value is determined by the network type. For broadcast networks and P2P networks, the default value is 10s. For P2MP networks and NBMA networks, the default value is 30s.
Configure the OSPFv3 neighbor dead time.	**ipv6 ospf dead-interval** *interval-value* [**instance-id** *instance-id*]	Optional. By default, the dead time is four times of the Hello interval.

Note:

- The Hello interval and neighbor dead time of OSPFv3 neighbors must be the same; otherwise, they cannot set up neighbor relations.
- When you modify the Hello interval, if the current neighbor dead time is four times of the Hello interval, the neighbor dead time is automatically modified to be still four times of the new Hello interval. If the current neighbor dead time is not four times of the Hello interval, the neighbor dead time keeps unchanged.
- If you modify the neighbor dead time, the Hello interval is not affected.

Configure an OSPFv3 Passive Interface

The dynamic routing protocol adopts a passive interface to effectively decrease the network bandwidth consumed by the routing protocol. After an OSPFv3 passive interface is configured, you can use the **enable** command of the interface to advertise the routes of the directly connected network segment in which the interface is located, but the receiving and transmitting of OSPFv3 packets are damped on the interface.

Table 7-24 Configure an OSPFv3 passive interface

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the OSPFv3 configuration mode.	**ipv6 router ospf** *process-id* [**vrf** *vrf-name*]	-
Configure an OSPFv3 passive interface.	**passive-interface** {*interface-name*|**default**}	Mandatory. By default, no OSPFv3 passive interface is configured.

Configure an OSPFv3 Demand Circuit

On P2P and P2MP links, to decrease the line cost, you can configure an OSPFv3 demand circuit to suppress periodical transmitting of Hello packets and periodical update of LSA packets. This function is mainly applied on charged links such as ISDN, SVC, and X.25.

Table 7-25 Configure an OSPFv3 demand circuit

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the interface configuration mode.	**interface** *interface-name*	-
Configure an OSPFv3 demand circuit.	**ipv6 ospf demand-circuit** [**instance-id** *instance-id*]	Mandatory. By default, no OSPFv3 demand circuit is enabled.

Configure the Priority of an OSPFv3 Interface

Interface priorities are mainly used in election of Designated Router (DR), and Backup Designated Router (BDR) in broadcast networks and NBMA networks. The value range is 0-255. The larger the value is, the higher the priority is. The default value is 1.

The DR and BDR are selected from all devices in a network segment based on interface priorities and Router IDs through Hello packets. The rules are as follows:

- First, the device whose interface has the highest priority is elected as the DR, and the device whose interface has the second highest priority is elected as the BDR. The device whose interface has the priority 0 does not participate in the election.

- If the interface priorities of two devices are the same, the device with the largest Router ID is elected as the DR, and the device with the second largest Router ID is elected as the BDR.
- If the DR fails, the BDR becomes the DR immediately, and a new BDR is elected.

Table 7-26 Configure the priority of an OSPFv3 interface

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the interface configuration mode.	**interface** *interface-name*	-
Configure the priority of an OSPFv3 interface.	**ipv6 ospf priority** *priority-value* [**instance-id** *instance-id*]	Optional. By default, the OSPFv3 interface priority is 1.

Note:

- Interface priorities affect only an election process. If the DR and BDR have already been elected, modification of interface priorities does not affect the election result; instead, it affects the next election of DR or BDR. Therefore, the DR may not have the interface with the highest priority, and the BDR may not have the interface with the second highest priority.

Configure the OSPFv3 Interface to Ignore MTU

When adjacent OSPF devices exchange DD packets, MTUs are checked by default. If the MTUs are different, the devices cannot form a neighbor relation. If you have configured OSPFv3 to ignore interface MTU check, even if MTUs are different, they can set up a neighbor relation.

Table 7-27 Configure an OSPFv3 interface to ignore MTU

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the interface configuration mode.	**interface** *interface-name*	-
Configure the OSPFv3 interface to ignore MTU.	**ipv6 ospf mtu-ignore** [**instance-id** *instance-id*]	Mandatory By default, the OSPFv3 interface performs the MTU consistency check.

Configure the LSA Transmit Delay of an OSPFv3 Interface

LSA transmit delay refers to the time it takes for an LSA to flood to other devices. The device that sends the LSA adds the interface transmit delay to the LSA aging time. By default, once the flooding LSA passes a device, the aging time is increased by 1. You can configure the LSA transmit delay according to the network conditions. The value range is 1-840. LSA transmit delay is usually configured on low-speed links.

Table 7-28 Configure the LSA transmit delay of an OSPFv3 interface

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the interface configuration mode.	**interface** *interface-name*	-
Configure the LSA transmit delay of an OSPFv3 interface.	**ipv6 ospf transmit-delay** *delay-value* **instance-id** [*instance-id*]	Optional. By default, the LSA transmit delay is 1s.

Configure OSPFv3 LSA Retransmission

To ensure the reliability of data exchange, OSPFv3 adopts the acknowledgement mechanism. If an LSA floods on a device interface, the LSA is added into the retransmission list of the neighbor. If no acknowledgement message is received from the neighbor after the retransmission time times out, the LSA is retransmitted until an acknowledgement message is received.

Table 7-29 Configure OSPFv3 LSA retransmission

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the interface configuration mode.	**interface** *interface-name*	-
Configure the interval of OSPFv3 LSA retransmission.	**ipv6 ospf retransmit-interval** *interval-value* [**instance-id** *instance-id*]	Optional. By default, the retransmission interval is 5s.

Configure OSPFv3 SPF Calculation Time

If the OSPFv3 network topology changes, routes need to be re-calculated. When the network continues to change, frequent route calculation occupies a lot of system resources. You can adjust the SPF calculation time parameters to prevent frequent network changes from consuming too many system resources.

Table 7-30 Configure OSPFv3 SPF calculation time

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the OSPFv3 configuration mode.	**ipv6 router ospf** *process-id* [**vrf** *vrf-name*]	-
Configure OSPFv3 SPF calculation time.	**timers throttle spf** *delay-time hold-time max-time*	Optional. By default, *delay-time* is 5000ms, *hold-time* is 10000ms, and *max-time* is 10000ms.

Note:

- The parameter *delay-time* indicates the initial calculation delay, *hold-time* indicates the suppression time, and *max-time* indicates the maximum waiting time between two SPF calculations. If network changes are not frequent, you can shorten the continuous route calculation interval to *delay-time*. If network changes are frequent, you can adjust the parameters, increase the suppression time to *hold-time*×2^{n-2} (n is the number of route calculation trigger times), extend the waiting time based on the configured *hold-time* increment and the maximum value must not exceed *max-time*.

7.2.8. Configure OSPFv3 Fast Re-Routing

Configuration Conditions

Before configuring the OSPFv3 fast re-routing, first complete the following task:

- Enable the IPv6 forwarding function
- Enable the OSPFv3 protocol

Configure OSPFv3 Fast Re-routing

In OSPFv3 network, due to link or device failure, the packets passing through the failure point will be discarded or a loop will be generated. The traffic interruption caused by this will continue until the protocol re-converges, which often lasts for several seconds. In order to reduce the traffic interruption time, OSPFv3 fast rerouting can be configured. By applying the route map, the backup next hop can be set for the successfully matched route. Once the main link fails, the traffic passing through the failed link will be immediately switched to the backup link, so as to realize fast switching.

Table 7-31 Configure OSPFv3 fast re-routing

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the OSPFv3 configuration mode.	**ipv6 router ospf** *process-id* [**vrf** *vrf-name*]	-
Configure the OSPFv3 process to enable the static fast re-routing function	**fast-reroute route-map** *route-map-name*	Mandatory By default, do not enable the OSPFv3 static fast re-routing function.
Configure the OSPFv3 process to enable the dynamic fast re-routing function	**fast-reroute loop-free-alternate** [**route-map** *route-map-name*]	Mandatory By default, do not enable the OSPFv3 dynamic fast re-routing function.
Configure the OSPFv3 process to enable the pic function	**pic**	Mandatory After enabling the pic function, enable the auto fast re-routing function. By default, do not enable the OSPFv3 pic function.

Note:

- OSPFv3 fast re-routing function can be divided into static fast rerouting and dynamic fast rerouting.
- The static fast rerouting function needs to associate the route map, and set the next hop interface and address of the backup route in the route-map.
- At present, dynamic fast rerouting only supports point-to-point network, that is, the network type of all outgoing interfaces of the device needs to be point-to-point. After configuring dynamic fast rerouting, the device automatically calculates and sets the backup next hop interface and address. Dynamic fast rerouting can also be associated with route-map. Only the routes matching route map are set to back up the next hop interface and address.
- The various modes of enabling rerouting are mutually exclusive.

7.2.9. Configure OSPFv3 to Coordinate with BFD

Configuration Condition

Before configuring OSPFv3 to coordinate with BFD, ensure that:

- The IPv6 forwarding function is enabled.
- The OSPFv3 protocol is enabled.

Configure OSPFv3 to Coordinate with BFD

Bidirectional Forwarding Detection (BFD) provides a method for quickly detecting the status of a line between two devices. If BFD is started between two adjacent OSPFv3 devices, if the line between two devices becomes faulty, BFD quickly detects the fault and informs OSPFv3 of the fault. Then, it triggers OSPFv3 to start route calculation and switch over to the backup line, achieving fast switchover of routes.

Table 7-32 Configure OSPFv3 to coordinate with BFD

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the interface configuration mode.	**interface** *interface-name*	-
Enable or disable BFD on the specified OSPFv3 interface.	**ipv6 ospf bfd [disable] [instance-id** *instance-id* **]**	Mandatory. By default, the BFD function is disabled.
Enter the global configuration mode.	**exit**	-
Enter the OSPFv3 configuration mode.	**ipv6 router ospf** *process-id* [**vrf** *vrf-name*]	-
Enable BFD on all interfaces of the OSPFv3 process.	**bfd all-interfaces**	Optional

Note:

- If BFD is configured both in OSPFv3 configuration mode and interface configuration mode, the BFD configuration in the interface has the higher priority.

7.2.10. OSPFv3 Monitoring and Maintaining

Table 7-33 OSPFv3 monitoring and maintaining

Command	Description
clear ipv6 ospf err-statistic	Clear the OSPFv3 error statistics information.
clear ipv6 ospf [*process-id*] **process**	Reset an OSPFv3 process.
clear ipv6 ospf [*process-id*] **redistribution**	Re-advertise external routes.
clear ipv6 ospf [*process-id*] **route**	Re-calculate OSPFv3 routes.
clear ipv6 ospf statistics [*interface-name*]	Clear the OSPFv3 interface statistics information
show ipv6 ospf [*process-id*]	Display the OSPFv3 basic information.
show ipv6 ospf [*process-id*] **border-routers**	Display the information about the routes to the boundary devices in OSPFv3.
show ipv6 ospf core-info	Display the core information of the OSPFv3 process
show ipv6 ospf [*process-id*] **database** [**database-summary** | **external | inter-prefix** | **inter-router** | **intra-prefix** | **link** | **network** | **nssa-external | grace** | **router** | **adv-router** *router-id* | **age** *lsa_age* | **max-age | self-originate**]	Display the information about an OSPFv3 database.
show ipv6 ospf error-statistic	Display the OSPFv3 error statistics information
show ipv6 ospf event-list	Display the receiving queue information of the OSPFv3 packet
show ipv6 ospf interface [*interface-name* [**detail**]]	Display the information about an OSPFv3 interface.

Command	Description
show ipv6 ospf [*process-id*] **neighbor** [*neighbor-id* | **all** | **detail** [**all**] | **interface** *interface-name* [**detail**] | **statistics**]	Display the information about OSPFv3 neighbors.
show ipv6 ospf [*process-id*] **route** [*ipv6-prefix/prefix-length* | **connected** | **external** | **inter-area** | **intra-area** | **statistic**]	Display the information about OSPFv3 routes.
show ipv6 ospf [*process-id*] **sham-link**	Display the configured OSPFv3 sham link information about, including interface status, cost value, and neighbor status.
show ipv6 ospf [*process-id*] **topology area** [*area-id*]	Display the OSPFv3 topology information
show ipv6 ospf [*process- id*] **virtual-links**	Display the OSPFv3 virtual link information
show ipv6 ospf [**vrf** *vrf-name*]	Display the all OSPFv3 process information and parameters in the specified vrf
show running-config ipv6 router ospf	Display the OSPFv3 running configuration

7.3. OSPFv3 Typical Configuration Example

7.3.1. Configure OSPFv3 Basic Functions

Network Requirements

- Configure the OSPFv3 protocol for all devices, and divide the devices into three areas: Area 0, Area 1, and Area 2. After configuration, all devices should be able to learn routes from each other.
- On a back-to-back Ethernet interface, to speed up set of OSPFv3 neighbors, you can change the network type of the OSPFv3 interface to P2P. Modify the network type of the interfaces in Area 2 to P2P. After the configuration, all devices can learn routes from each other.

Network Topology

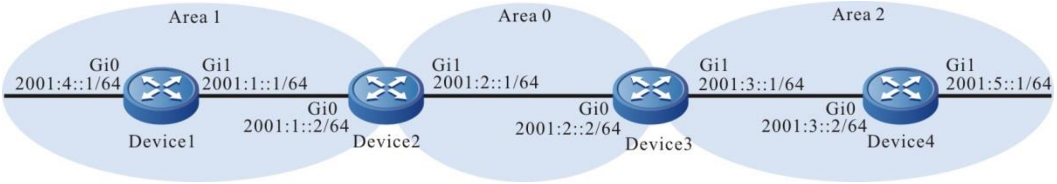

Figure 7-1 Networking for configuring basic OSPFv3 functions

Configuration Steps

Step 1: Configure the IPv6 addresses of the interfaces. (Omitted)

Step 2: Configure an OSPFv3 process and let the interface cover different areas.

#On Device1, configure an OSPFv3 process and configure the interfaces to cover area 1.

```
Device1#configure terminal
Device1(config)#ipv6 router ospf 100
Device1(config-ospf6)#router-id 1.1.1.1
Device1(config-ospf6)#exit
Device1(config)#interface gigabitethernet0
Device1(config-if-gigabitethernet0)#ipv6 router ospf 100 area 1
Device1(config-if-gigabitethernet0)#exit
Device1(config)#interface gigabitethernet1
Device1(config-if-gigabitethernet1)#ipv6 router ospf 100 area 1
Device1(config-if-gigabitethernet1)#exit
```

#On Device2, configure an OSPFv3 process and configure the interfaces to cover Area 0 and Area 1.

```
Device2#configure terminal
Device2(config)#ipv6 router ospf 100
Device2(config-ospf6)#router-id 2.2.2.2
Device2(config-ospf6)#exit
Device2(config)#interface gigabitethernet0
Device2(config-if-gigabitethernet0)#ipv6 router ospf 100 area 1
Device2(config-if-gigabitethernet0)#exit
Device2(config)#interface gigabitethernet1
Device2(config-if-gigabitethernet1)#ipv6 router ospf 100 area 0
Device2(config-if-gigabitethernet1)#exit
```

#On Device3, configure an OSPFv3 process and configure the interfaces to cover Area 0 and Area 2.

```
Device3#configure terminal
Device3(config)#ipv6 router ospf 100
Device3(config-ospf6)#router-id 3.3.3.3
```

```
Device3(config-ospf6)#exit
Device3(config)#interface gigabitethernet0
Device3(config-if-gigabitethernet0)#ipv6 router ospf 100 area 0
Device3(config-if-gigabitethernet0)#exit
Device3(config)#interface gigabitethernet1
Device3(config-if-gigabitethernet1)#ipv6 router ospf 100 area 2
Device3(config-if-gigabitethernet1)#exit
```

#On Device4, configure an OSPFv3 process and configure the interfaces to cover area 2.

```
Device4#configure terminal
Device4(config)#ipv6 router ospf 100
Device4(config-ospf6)#router-id 4.4.4.4
Device4(config-ospf6)#exit
Device4(config)#interface gigabitethernet0
Device4(config-if-gigabitethernet0)#ipv6 router ospf 100 area 2
Device4(config-if-gigabitethernet0)#exit
Device4(config)#interface gigabitethernet1
Device4(config-if-gigabitethernet1)#ipv6 router ospf 100 area 2
Device4(config-if-gigabitethernet1)#exit
```

Note:

- The Router ID in OSFPv3 must be configured manually, and the Router IDs of any two routers in the AS cannot be the same.
- When an interface is enabled to OSPFv3, it is necessary to specify which interface instance is enabled to the OSPFv3 process, and the two instance numbers should be consistent. By default, it is in instance 0.

#Query the OSPFv3 neighbor information and routing table of Device1.

```
Device1#show ipv6 ospf neighbor
OSPFv3 Process (100)
Neighbor ID    Pri   State        Dead Time   Interface              Instance ID
2.2.2.2          1   Full/DR       00:00:38    gigabitethernet1            0

Device1#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management

L    ::1/128 [0/0]
     via ::, 00:41:07, lo0
C    2001:1::/64 [0/0]
```

```
    via ::, 00:32:19, gigabitethernet1
L   2001:1::1/128 [0/0]
    via ::, 00:32:18, lo0
O   2001:2::/64 [110/2]
    via fe80::201:7aff:fe5e:6d45, 00:23:06, gigabitethernet1
O   2001:3::/64 [110/3]
    via fe80::201:7aff:fe5e:6d45, 00:23:00, gigabitethernet1
C   2001:4::/64 [0/0]
    via ::, 00:16:46, gigabitethernet0
L   2001:4::1/128 [0/0]
    via ::, 00:16:45, lo0
O   2001:5::/64 [110/4]
    via fe80::201:7aff:fe5e:6d45, 00:01:42, gigabitethernet1
```

#Query the OSPFv3 neighbors and routing table of Device2.

```
Device2#show ipv6 ospf neighbor
OSPFv3 Process (100)
Neighbor ID     Pri   State          Dead Time   Interface                  Instance ID
1.1.1.1          1   Full/Backup    00:00:34    gigabitethernet0              0
3.3.3.3          1   Full/DR        00:00:33    gigabitethernet1              0

Device2#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management

L   ::1/128 [0/0]
    via ::, 00:50:36, lo0
C   2001:1::/64 [0/0]
    via ::, 00:43:05, gigabitethernet0
L   2001:1::2/128 [0/0]
    via ::, 00:43:04, lo0
C   2001:2::/64 [0/0]
    via ::, 00:40:01, gigabitethernet1
L   2001:2::1/128 [0/0]
    via ::, 00:39:57, lo0
O   2001:3::/64 [110/2]
    via fe80::2212:1ff:fe01:101, 00:34:00, gigabitethernet1
```

```
O   2001:4::/64 [110/2]
    via fe80::201:7aff:fe61:7a24, 00:27:28, gigabitethernet0
O   2001:5::/64 [110/3]
    via fe80::2212:1ff:fe01:101, 00:12:41, gigabitethernet1
```

#Query OSPFv3 Link Status Database (LSDB) of Device2.

```
Device2#show ipv6 ospf database

            OSPFv3 Router with ID (2.2.2.2) (Process 100)

                Link-LSA (Interface gigabitethernet0)
Link State ID   ADV Router      Age Seq#       CkSum  Prefix
0.0.0.1         1.1.1.1          81 0x80000001 0x8d18      1
0.0.0.1         2.2.2.2          78 0x80000001 0xf996      1

                Link-LSA (Interface gigabitethernet1)
Link State ID   ADV Router      Age Seq#       CkSum  Prefix
0.0.0.2         2.2.2.2          71 0x80000003 0x2467      1
0.0.0.1         3.3.3.3          35 0x80000003 0xcd12      1

                Router-LSA (Area 0.0.0.0)
Link State ID   ADV Router      Age Seq#       CkSum    Link
0.0.0.0         2.2.2.2          37 0x80000004 0x0dd6      1
0.0.0.0         3.3.3.3          25 0x80000007 0xda03      1

                Network-LSA (Area 0.0.0.0)
Link State ID   ADV Router      Age Seq#       CkSum
0.0.0.1         3.3.3.3          35 0x80000001 0x5790

                Inter-Area-Prefix-LSA (Area 0.0.0.0)
Link State ID   ADV Router      Age Seq#       CkSum  Prefix
0.0.0.2         2.2.2.2          42 0x80000007 0x9e25 2001:1::/64
0.0.0.3         2.2.2.2          23 0x80000002 0xcef4 2001:4::/64
0.0.0.1         3.3.3.3          35 0x80000005 0xaa16 2001:3::/64
0.0.0.3         3.3.3.3          55 0x80000001 0xc0fe 2001:5::/64

                Intra-Area-Prefix-LSA (Area 0.0.0.0)
Link State ID   ADV Router      Age Seq#       CkSum  Prefix  Reference
```

```
0.0.0.3        3.3.3.3          34 0x80000001 0xb2d3     1  Network-LSA

               Router-LSA (Area 0.0.0.1)
Link State ID   ADV Router       Age Seq#       CkSum    Link
0.0.0.0        1.1.1.1          41 0x80000004 0xc726      1
0.0.0.0        2.2.2.2          37 0x80000004 0xac3c      1

               Network-LSA (Area 0.0.0.1)
Link State ID   ADV Router       Age Seq#       CkSum
0.0.0.1        2.2.2.2          42 0x80000001 0x21d2

               Inter-Area-Prefix-LSA (Area 0.0.0.1)
Link State ID   ADV Router       Age Seq#       CkSum  Prefix
0.0.0.1        2.2.2.2          42 0x80000004 0xbc0a 2001:2::/64
0.0.0.4        2.2.2.2          19 0x80000001 0xb80c 2001:3::/64
0.0.0.5        2.2.2.2          19 0x80000001 0xd0ef 2001:5::/64

               Intra-Area-Prefix-LSA (Area 0.0.0.1)
Link State ID   ADV Router       Age Seq#       CkSum  Prefix  Reference
0.0.0.1        1.1.1.1          35 0x80000005 0xc4ce      1  Router-LSA
0.0.0.3        2.2.2.2          41 0x80000001 0x8807      1  Network-LSA
```

For Device2, routes 2001:3::/642 and 2001:5::/64 are inter-area routes. You can query the LSA information of the related routes in Inter-Area-Prefix-LSA (Area 0.0.0.0). In the case of intra-area routes, run the **show ipv6 ospf database intra-prefix** command to query the LSA information of the related routes.

Step 3: Configure the network type of OSPFv3 interfaces to P2P.

#On Device3, configure the OSPFv3 network type of interface gigabitethernet1 to P2P.

```
Device3(config)#interface gigabitethernet1
Device3(config-if-gigabitethernet1)#ipv6 ospf network point-to-point
Device3(config-if-gigabitethernet1)#exit
```

#On Device4, configure the OSPFv3 network type of interface gigabitethernet0 to P2P.

```
Device4(config)#interface gigabitethernet0
Device4(config-if-gigabitethernet0)#ipv6 ospf network point-to-point
Device4(config-if-gigabitethernet0)#exit
```

Step 4: Check the result.

#Query the OSPFv3 neighbors and routing table of Device3.

```
Device3#show ipv6 ospf neighbor
```

```
OSPFv3 Process (100)
Neighbor ID    Pri   State          Dead Time   Interface                    Instance ID
2.2.2.2          1   Full/Backup    00:00:39   gigabitethernet0          0
4.4.4.4          1   Full/ -        00:00:39   gigabitethernet1          0

Device3#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management

L   ::1/128 [0/0]
    via ::, 1d:09:10:10, lo0
O   2001:1::/64 [110/2]
    via fe80::201:7aff:fe5e:6d46, 02:07:25, gigabitethernet0
C   2001:2::/64 [0/0]
    via ::, 03:07:51, gigabitethernet0
L   2001:2::2/128 [0/0]
    via ::, 03:07:48, lo0
C   2001:3::/64 [0/0]
    via ::, 03:07:41, gigabitethernet1
L   2001:3::1/128 [0/0]
    via ::, 03:07:39, lo0
O   2001:4::/64 [110/3]
    via fe80::201:7aff:fe5e:6d46, 02:07:25, gigabitethernet0
O   2001:5::/64 [110/2]
    via fe80::201:2ff:fe03:405, 00:00:22, gigabitethernet1
```

Note:

- If OSPFv3 neighbor relations are set up in a P2P network, no DR or BDR election will be performed.

#Query the OSPFv3 neighbors and routing table of Device4.

```
Device4#show ipv6 ospf neighbor
OSPFv3 Process (100)
Neighbor ID    Pri   State          Dead Time   Interface                    Instance ID
3.3.3.3          1   Full/ -        00:00:38   gigabitethernet0             0

Device4#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
```

```
      U - Per-user Static route
      O - OSPF, OE-OSPF External, M - Management

L   ::1/128 [0/0]
    via ::, 00:05:34, lo0
O   2001:1::/64 [110/3]
    via fe80::2212:1ff:fe01:102, 00:03:12, gigabitethernet0
O   2001:2::/64 [110/2]
    via fe80::2212:1ff:fe01:102, 00:03:12, gigabitethernet0
C   2001:3::/64 [0/0]
    via ::, 00:04:34, gigabitethernet0
L   2001:3::2/128 [0/0]
    via ::, 00:04:31, lo0
O   2001:4::/64 [110/4]
    via fe80::2212:1ff:fe01:102, 00:03:12, gigabitethernet0
C   2001:5::/64 [0/0]
    via ::, 00:03:14, gigabitethernet1
L   2001:5::1/128 [0/0]
    via ::, 00:03:13, lo0
```

After the network type of OSPFv3 interfaces are modified to P2P, neighbors can be set up normally, and routes can be learned normally.

7.3.2. Configure OSPFv3 to Use IPSec Encryption Authentication

Network Requirements

- All routers run OSPFv3, and the whole AS is divided to two areas.
- Device1, Device2, and Device3 use the IPsec tunnel to perform the encryption authentication for the OSPFv3 protocol packets; Device1 and Device2 adopt the ESP transmission encapsulation mode, the encryption algorithm is 3des, and the authentication algorithm is sha1; Device2 and Device3 adopt the ESP transmission encapsulation mode, the encryption algorithm is aes128, and the ESP authentication algorithm is sm3.
- After configuration, the device can normally set up the neighbor and lear the routes from each other.

Network Topology

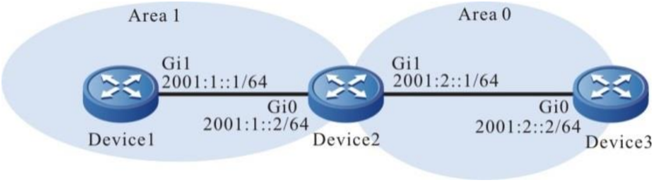

Figure 7-2 Networking for configuring OSPFv3 to use the IPSec encryption authentication

Configuration Steps

Step 1: Configure the IPv6 addresses of the interfaces. (Omitted)

Step 2: Configure an OSPFv3 process, and enable the OSPFv3 function on the corresponding interface.

#Configure the OSPFv3 process of Device1, Device2, and Device3, and enable OSPFv3 on the interface.

```
Device1#configure terminal
Device1(config)#ipv6 router ospf 100
Device1(config-ospf6)#router-id 1.1.1.1
Device1(config-ospf6)#exit
Device1(config)#interface gigabitethernet0
Device1(config-if-gigabitethernet0)#ipv6 router ospf 100 area 1
Device1(config-if-gigabitethernet0)#exit

Device2#configure terminal
Device2(config)#ipv6 router ospf 100
Device2(config-ospf6)#router-id 2.2.2.2
Device2(config-ospf6)#exit
Device2(config)#interface gigabitethernet0
Device2(config-if-gigabitethernet0)#ipv6 router ospf 100 area 1
Device2(config-if-gigabitethernet0)#exit
Device2(config)#interface gigabitethernet1
Device2(config-if-gigabitethernet1)#ipv6 router ospf 100 area 0
Device2(config-if-gigabitethernet1)#exit

Device3#configure terminal
Device3(config)#ipv6 router ospf 100
Device3(config-ospf6)#router-id 3.3.3.3
Device3(config-ospf6)#exit
Device3(config)#interface gigabitethernet0
Device3(config-if-gigabitethernet0)#ipv6 router ospf 100 area 0
Device3(config-if-gigabitethernet0)#exit
```

Step 3: Configure the IPSec proposal and manual tunnel.

#Configure Device1, create the IPSec proposal a, adopt the ESP transmission encapsulation mode, encryption algorithm 3des, and authentication algorithm sha1, create IPSec manual tunnel a, and configure the SPI and key.

```
Device1(config)#crypto ipsec proposal a
```

```
Device1(config-ipsec-prop)#mode transport
Device1(config-ipsec-prop)#esp 3des sha1
Device1(config-ipsec-prop)#exit
Device1(config)#crypto ipv6-tunnel a manual
Device1(config-manual-tunnel)#set ipsec proposal a
Device1(config-manual-tunnel)#set inbound esp 1000 encryption 0 111111111111111111111111 authentication 0 aaaaaaaaaaaaaaaaaaaa
Device1(config-manual-tunnel)#set outbound esp 1001 encryption 0 aaaaaaaaaaaaaaaaaaaaaaaa authentication 0 11111111111111111111
Device1(config-manual-tunnel)#exit
```

#Configure Device2, create the IPSec proposal a, adopt the ESP transmission encapsulation mode, encryption algorithm 3des, and authentication algorithm sha1, create IPSec manual tunnel a, and configure the SPI and key ; create IPSec proposal b, adopt the ESP transmission encapsulation mode, encryption algorithm aes128, and authentication algorithm sm3, create IPSec manual tunnel b, and configure the SPI and key.

```
Device2(config)#crypto ipsec proposal a
Device2(config-ipsec-prop)#mode transport
Device2(config-ipsec-prop)#esp 3des sha1
Device2(config-ipsec-prop)#exit
Device2(config)#crypto ipv6-tunnel a manual
Device2(config-manual-tunnel)#set ipsec proposal a
Device2(config-manual-tunnel)#set inbound esp 1001 encryption 0 aaaaaaaaaaaaaaaaaaaaaaaa authentication 0 11111111111111111111
Device2(config-manual-tunnel)#set outbound esp 1000 encryption 0 111111111111111111111111 authentication 0 aaaaaaaaaaaaaaaaaaaa
Device2(config-manual-tunnel)#exit
Device2(config)#crypto ipsec proposal b
Device2(config-ipsec-prop)#mode transport
Device2(config-ipsec-prop)#esp aes128 sm3
Device2(config-ipsec-prop)#exit
Device2(config)#crypto ipv6-tunnel b manual
Device2(config-manual-tunnel)#set ipsec proposal b
Device2(config-manual-tunnel)#set inbound esp 2001 encryption 0 1111111111111111 authentication 0 aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
Device2(config-manual-tunnel)#set outbound esp 2000 encryption 0 1111111111111111 authentication 0 11111111111111111111111111111111
Device2(config-manual-tunnel)#exit
```

#Configure Device3, create the IPSec proposal b, adopt the ESP transmission encapsulation mode, encryption algorithm aes128, and authentication algorithm sm3, create IPSec manual tunnel b, and configure the SPI and key.

```
Device3(config)#crypto ipsec proposal b
Device3(config-ipsec-prop)#mode transport
Device3(config-ipsec-prop)#esp aes128 sm3
Device3(config-ipsec-prop)#exit
Device3(config)#crypto ipv6-tunnel b manual
Device3(config-manual-tunnel)#set ipsec proposal b
Device3(config-manual-tunnel)#set inbound esp 2000 encryption 0 1111111111111111
authentication 0 11111111111111111111111111111111
Device3(config-manual-tunnel)#set outbound esp 2001 encryption 0 1111111111111111
authentication 0 aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
Device3(config-manual-tunnel)#exit
```

Step 4: In the OSPFv3 process, the areas are bound to the corresponding IPSec tunnel.

#In the OSPFv3 process of Device1, area 1 is bound to IPSec tunnel a.

```
Device1(config)#ipv6 router ospf 100
Device1(config-ospf6)#area 1 ipsec-tunnel a
Device1(config-ospf6)#exit
```

#In the OSPFv3 process of Device2, area 1 is bound to IPSec tunnel a, and area 0 is bound to IPSec tunnel b.

```
Device2(config)#ipv6 router ospf 100
Device2(config-ospf6)#area 1 ipsec-tunnel a
Device2(config-ospf6)#area 0 ipsec-tunnel b
Device1(config-ospf6)#exit
```

#In the OSPFv3 process of Device3, area 0 is bound to IPSec tunnel b.

```
Device3(config)#ipv6 router ospf 100
Device3(config-ospf6)#area 0 ipsec-tunnel b
Device3(config-ospf6)#exit
```

Step 5: Check the result.

#View the OSPFv3 process information of Device1.

```
Device1#show ipv6 ospf 100
Routing Process "OSPFv3 (100)" with ID 1.1.1.1
 Process bound to VRF default
 IETF graceful-restarter support disabled
 IETF gr helper support enabled
 Initial SPF schedule delay 5000 msecs
 Minimum hold time between two consecutive SPFs 10000 msecs
 Maximum wait time between two consecutive SPFs 10000 msecs
 Minimum LSA interval 5 secs, Minimum LSA arrival 1 secs
```

```
Number of external LSA 0. Checksum Sum 0x0000
Number of AS-Scoped Unknown LSA 0
Number of LSA originated 5
Number of LSA received 5
Number of areas in this router is 1
Not Support Demand Circuit lsa number is 0
Autonomy system support flood DoNotAge Lsa
  Area 0.0.0.1
     Number of interfaces in this area is 1
     IPSec Tunnel Name:a , ID: 154
     Number of fully adjacent neighbors in this area is 1
     Number of fully adjacent sham-link neighbors in this area is 0
     Number of fully adjacent virtual neighbors through this area is 0
     SPF algorithm executed 4 times
     LSA walker due in 00:00:02
     Number of LSA 4.  Checksum Sum 0x2FC53
     Number of Unknown LSA 0
     Not Support Demand Circuit lsa number is 0
     Indication lsa (by other routers) number is: 0,
     area support flood DoNotAge Lsa
```

You can see that the area is bound to IPSec tunnel a, and the ID is a random value of 0-1023.

#View the IPSec tunnel information of Device1.

```
Device1#show crypto ipv6-tunnel a
get the manual tunnel
Crypto tunnel a : MANUAL
      policy name : (null)
      peer address :
      local interface : (null) address :
      Ipsec proposal : a
      Inbound :
        esp : spi: 1000 encription key: ******** authentication key: ********
        ah spi: 0 authentication key: (null)
      Outbound :
        esp spi: 1001 encryption key: ******** authentication key: ********
        ah spi: 0 authentication key: (null)
      route ref : 1
      route asyn : 1
```

```
    route rt_id : 154
```

You can see that route rt_id is equal to the ID in show ipv6 ospf 100.

#View the IPSec tunnel encryption type information of Device1.

```
Device1#show crypto ipsec sa tunnel a
route policy:
 the pairs of ESP ipsec sa : id :0 , algorithm : 3DES HMAC-SHA1-96
  inbound esp ipsec sa : spi : 0x3e8(1000) crypto m_context(s_context) : 0x4cd3ba78 / 0x4cd3bae0
    current input 26 packets, 2 kbytes
    encapsulation mode : Transport
    replay protection : OFF
    remaining lifetime (seconds/kbytes) : 0/0
    uptime is 0 hour 4 minute 45 second
  outbound esp ipsec sa : spi : 0x3e9(1001) crypto m_context(s_context) : 0x4cd3bb48 / 0x4cd3bbb0
    current output 39 packets, 3 kbytes
    encapsulation mode : Transport
    replay protection : OFF
    remaining lifetime (seconds/kbytes) : 0/0
    uptime is 0 hour 4 minute 45 second

total sa and sa group is 1
```

You can see that IPSec tunnel a adopts the ESP transmission encapsulation mode, encryption algorithm 3des, and authentication algorithm sha1.

#View the OSPFv3 interface information of Device1.

```
Device1#show ipv6 ospf interface gigabitethernet0
gigabitethernet0 is up, line protocol is up
 Interface ID 50331913
 IPv6 Prefixes
  fe80::201:7aff:fecf:fbec/10 (Link-Local Address)
  2001 :1::1/64
 Interface ID 13
 OSPFv3 Process (100), Area 0.0.0.1, Instance ID 0, Enabled 00:41:10, MTU 1500
  Router ID 1.1.1.1, Network Type BROADCAST, Cost: 1
  IPSec tunnel(Area):a, ID:154
  Transmit Delay is 1 sec, State Backup, 3 state change, Priority 1
  Designated Router (ID) 2.2.2.2
   Interface Address fe80::200:1ff:fe7a:adf0
  Backup Designated Router (ID) 1.1.1.1
   Interface Address fe80::201:7aff:fecf:fbec
  Timer interval configured, Hello 10, Dead 39, Wait 39, Retransmit 5
   Hello due in 00:00:06
  Neighbor Count is 1, Adjacent neighbor count is 1
 Hello received 2 sent 3, DD received 3 sent 4
 LS-Req received 1 sent 1, LS-Upd received 5 sent 3
 LS-Ack received 3 sent 2, Discarded 0
```

You can see that the interface is bound to IPSec tunnel a, and the ID is a random value of 0-1023.

#View the OSPFv3 neighbor information and core route table of Device1.

```
Device1#show ipv6 ospf neighbor
OSPFv3 Process (100)
Neighbor ID    Pri    State       Dead Time      Interface             Instance ID
2.2.2.2        1      Full/DR     00:00:39       gigabitethernet0    0

Device1#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management

L   ::1/128 [0/0]
    via ::, 4d:04:06:36, lo0
C   2001:1::/64 [0/0]
    via ::, 03:00:53, gigabitethernet0
L   2001:1::1/128 [0/0]
    via ::, 03:00:49, lo0
O   2001:2::/64 [110/2]
    via fe80::201:7aff:fec9:1cdd, 2d:00:03:49, gigabitethernet0
```

On Device1, the neighbor is normally set up, and the route is learnt normally.

#View the OSPFv3 process information of Device3.

```
Device3#show ipv6 ospf 100
 Routing Process "OSPFv3 (100)" with ID 3.3.3.3
 Process bound to VRF default
 IETF graceful-restarter support disabled
 IETF gr helper support enabled
 Initial SPF schedule delay 5000 msecs
```

```
Minimum hold time between two consecutive SPFs 10000 msecs
Maximum wait time between two consecutive SPFs 10000 msecs
Minimum LSA interval 5 secs, Minimum LSA arrival 1 secs
Number of external LSA 0. Checksum Sum 0x0000
Number of AS-Scoped Unknown LSA 0
Number of LSA originated 5
Number of LSA received 6
Number of areas in this router is 1
Not Support Demand Circuit lsa number is 0
Autonomy system support flood DoNotAge Lsa
  Area BACKBONE(0)
     Number of interfaces in this area is 1
     IPSec Tunnel Name:b , ID: 2
     Number of fully adjacent neighbors in this area is 1
     Number of fully adjacent sham-link neighbors in this area is 0
     SPF algorithm executed 4 times
     LSA walker due in 00:00:02
     Number of LSA 4.  Checksum Sum 0x24272
     Number of Unknown LSA 0
     Not Support Demand Circuit lsa number is 0
     Indication lsa (by other routers) number is: 0,
     area support flood DoNotAge Lsa
```

You can see that the area is bound to IPSec tunnel b, and the ID is a random value of 0-1023.

#View the IPSec tunnel information of Device3.

```
Device3#show crypto ipv6-tunnel b
get the manual tunnel
Crypto tunnel b : MANUAL
      policy name : (null)
      peer address :
      local interface : (null) address :
      Ipsec proposal : b
      Inbound :
        esp : spi: 2000 encription key: ******** authentication key: ********
        ah spi: 0 authentication key: (null)
      Outbound :
        esp spi: 2001 encryption key: ******** authentication key: ********
        ah spi: 0 authentication key: (null)
      route ref : 1
      route asyn : 1
      route rt_id : 2
```

You can see that route rt_id is equal to the ID in show ipv6 ospf 100.

#View the IPSec tunnel encryption type information of Device3.

```
Device3#show crypto ipsec sa tunnel b
route policy:
  the pairs of ESP ipsec sa : id : 0, algorithm : AES128 HMAC-SM3
inbound esp ipsec sa :  spi : 0x7d0(2000)  crypto m_context(s_context) : 0x6a0d9a98
/
0x6a0d9a30
        current input 53 packets, 5 kbytes
        encapsulation mode : Transport
        replay protection : OFF
        remaining lifetime (seconds/kbytes) : 0/0
        uptime is 0 hour 6 minute 40 second
outbound esp ipsec sa :  spi : 0x7d1(2001)  crypto m_context(s_context) : 0x6a0d99c8
/
0x6a0d9960
        current output 52 packets, 5 kbytes
        encapsulation mode : Transport
        replay protection : OFF
        remaining lifetime (seconds/kbytes) : 0/0
        uptime is 0 hour 6 minute 40 second

total sa and sa group is 1
```

You can see that the IPSec tunnel adopts the ESP transmission encapsulation mode, encryption algorithm aes128, and authentication algorithm sm3.

#View the OSPFv3 interface information of Device3.

```
Device3#show ipv6 ospf interface gigabitethernet0
igabitethernet0 is up, line protocol is up
  Interface ID 50331899
  IPv6 Prefixes
    fe80::200:1ff:fe7a:adf0/10 (Link-Local Address)
    2001 :2::1/64
  Interface ID 9
  OSPFv3 Process (100), Area 0.0.0.1, Instance ID 0, Enabled 00:50:39, MTU 1500
    Router ID 3.3.3.3, Network Type BROADCAST, Cost: 1
    IPSec tunnel(Area):b, ID:2
    Transmit Delay is 1 sec, State DR, 4 state change, Priority 1
    Designated Router (ID) 2.2.2.2
      Interface Address fe80::200:1ff:fe7a:adf0
    Backup Designated Router (ID) 1.1.1.1
      Interface Address fe80::201:7aff:fecf:fbec
    Timer interval configured, Hello 10, Dead 39, Wait 39, Retransmit 5
      Hello due in 00:00:02
    Neighbor Count is 1, Adjacent neighbor count is 1
  Hello received 272 sent 316, DD received 12 sent 9
  LS-Req received 3 sent 5, LS-Upd received 19 sent 18
  LS-Ack received 11 sent 13, Discarded 0
```

You can see that the interface is bound with the IPsec tunnel b, and the ID is a random value of 0-1023.

#View the OSPFv3 neighbor information and the core route table of Device3.

```
Device3#show ipv6 ospf neighbor
OSPFv3 Process (100)
Neighbor ID    Pri    State        Dead Time       Interface            Instance ID
2.2.2.2        1      Full/Backup   00:00:35       gigabitethernet0    0

Device3#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M – Management

L   ::1/128 [0/0]
    via ::, 09:53:53, lo0
O   2001:1::/64 [110/2]
    via fe80::ae9c:e4ff:fe77:889e, 00:23:36, gigabitethernet0
C   2001:2::/64 [0/0]
    via ::, 03:05:16, gigabitethernet0
L   2001:2::2/128 [0/0]
    via ::, 03:05:13, lo0
```

On Device3, the neighbor is set up normally, and the route is learnt normally.

Step 6: Bind the OSPFv3 interface with the corresponding IPSec tunnel.

#Configure Device1, and bind the interface gigabitethernet0 with IPSec tunnel a.

```
Device1(config)#interface gigabitethernet0
```

```
Device1(config-if-gigabitethernet0)#ipv6 ospf ipsec-tunnel a
Device1(config-if-gigabitethernet0)#exit
```

#Configure Device2, and bind the interface gigabitethernet1 with IPSec tunnel a; bind the interface gigabitethernet 2 with IPSec tunnel b.

```
Device2(config)#interface gigabitethernet0
Device2(config-if-gigabitethernet0)#ipv6 ospf ipsec-tunnel a
Device2(config-if-gigabitethernet0)#exit
Device2(config)#interface gigabitethernet1
Device2(config-if-gigabitethernet1)#ipv6 ospf ipsec-tunnel b
Device2(config-if-gigabitethernet1)#exit
```

#Configure Device3, and bind the interface gigabitethernet 0/2/3 with IPSec tunnel b.

```
Device3(config)#interface gigabitethernet0
Device3(config-if-gigabitethernet0)#ipv6 ospf ipsec-tunnel b
Device3(config-if-gigabitethernet0)#exit
```

Step 7: Check the result.

#View the OSPFv3 interface information of Device1.

```
Device1#show ipv6 ospf interface gigabitethernet0
gigabitethernet0 is up, line protocol is up
 Interface ID 50331913
 IPv6 Prefixes
  fe80::201:7aff:fecf:fbec/10 (Link-Local Address)
  2001 :1::1/64
 Interface ID 13
 OSPFv3 Process (100), Area 0.0.0.1, Instance ID 0, Enabled 00:41:10, MTU 1500
  Router ID 1.1.1.1, Network Type BROADCAST, Cost: 1
  IPSec tunnel:a, ID:154
  Transmit Delay is 1 sec, State Backup, 3 state change, Priority 1
  Designated Router (ID) 2.2.2.2
   Interface Address fe80::200:1ff:fe7a:adf0
  Backup Designated Router (ID) 1.1.1.1
   Interface Address fe80::201:7aff:fecf:fbec
  Timer interval configured, Hello 10, Dead 39, Wait 39, Retransmit 5
   Hello due in 00:00:06
  Neighbor Count is 1, Adjacent neighbor count is 1
 Hello received 2 sent 3, DD received 3 sent 4
 LS-Req received 1 sent 1, LS-Upd received 5 sent 3
 LS-Ack received 3 sent 2, Discarded 0
```

You can see that the interface is bound with the IPsec tunnel a, and the ID is a random value of 0-1023.

#View the OSPFv3 core route table of Device1.

```
Device1#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management

L   ::1/128 [0/0]
     via ::, 4d:04:06:36, lo0
C   2001:1::/64 [0/0]
     via ::, 03:00:53, gigabitethernet0
L   2001:1::1/128 [0/0]
     via ::, 03:00:49, lo0
O   2001:2::/64 [110/2]
     via fe80::201:7aff:fec9:1cdd, 2d:00:03:49, gigabitethernet0
```

On Device1, the route is learnt normally.

#View the OSPFv3 interface information of Device3.

```
Device3#show ipv6 ospf interface gigabitethernet0
igabitethernet0 is up, line protocol is up
  Interface ID 50331899
  IPv6 Prefixes
    fe80::200:1ff:fe7a:adf0/10 (Link-Local Address)
    2001 :2::1/64
  Interface ID 9
  OSPFv3 Process (100), Area 0.0.0.1, Instance ID 0, Enabled 00:50:39, MTU 1500
    Router ID 3.3.3.3, Network Type BROADCAST, Cost: 1
    IPSec tunnel:b, ID:2
    Transmit Delay is 1 sec, State DR, 4 state change, Priority 1
    Designated Router (ID) 2.2.2.2
      Interface Address fe80::200:1ff:fe7a:adf0
    Backup Designated Router (ID) 1.1.1.1
      Interface Address fe80::201:7aff:fecf:fbec
    Timer interval configured, Hello 10, Dead 39, Wait 39, Retransmit 5
      Hello due in 00:00:02
    Neighbor Count is 1, Adjacent neighbor count is 1
  Hello received 272 sent 316, DD received 12 sent 9
```

```
    LS-Req received 3 sent 5, LS-Upd received 19 sent 18
    LS-Ack received 11 sent 13, Discarded 0
```

You can see that the interface is bound with the IPsec tunnel b, and the ID is a random value of 0-1023.

#View the OSPFv3 core route table of Device3.

```
Device3#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M – Management

L   ::1/128 [0/0]
    via ::, 09:53:53, lo0
O   2001:1::/64 [110/2]
    via fe80::ae9c:e4ff:fe77:889e, 00:23:36, gigabitethernet0
C   2001:2::/64 [0/0]
    via ::, 03:05:16, gigabitethernet0
L   2001:2::2/128 [0/0]
    via ::, 03:05:13, lo0
```

On Device3, the route is learnt normally.

Note:

- When configuring OSPFv3 to bind the IPSec tunnel, you can only configure the area binding or interface binding, and also can configure the area and interface binding at the same time.
- When the area binding and interface binding configure the IPSec tunnel at the same time, the interface binding takes effect first.

7.3.3. Configure OSPFv3 to Coordinate with BFD

Network Requirements

- Configure OSPFv3 for all devices.
- Enable the BFD detection function on the line between Device1 and Device3. If the line becomes faulty, BFD quickly detects the fault and notify OSPFv3 of the fault. Then, OSPFv3 switches the route to Device2 for communication.

Network Topology

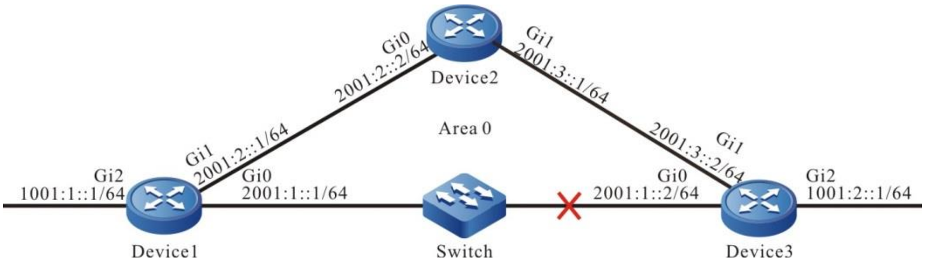

Figure 7-3 Networking for configuring OSPFv3 to coordinate with BFD

Configuration Steps

Step 1: Configure the IPv6 addresses of the interfaces. (Omitted)

Step 2: Configure an OSPFv3 process.

#Configure Device1.

```
Device1#configure terminal
Device1(config)#ipv6 router ospf 100
Device1(config-ospf6)#router-id 1.1.1.1
Device1(config-ospf6)#exit
Device1(config)#interface gigabitethernet0
Device1(config-if-gigabitethernet0)#ipv6 router ospf 100 area 0
Device1(config-if-gigabitethernet0)#exit
Device1(config)#interface gigabitethernet1
Device1(config-if-gigabitethernet1)#ipv6 router ospf 100 area 0
Device1(config-if-gigabitethernet1)#exit
Device1(config)#interface gigabitethernet2
Device1(config-if-gigabitethernet2)#ipv6 router ospf 100 area 0
Device1(config-if-gigabitethernet2)#exit
```

#Configure Device2.

```
Device2#configure terminal
Device2(config)#ipv6 router ospf 100
Device2(config-ospf6)#router-id 2.2.2.2
Device2(config-ospf6)#exit
Device2(config)#interface gigabitethernet0
Device2(config-if-gigabitethernet0)#ipv6 router ospf 100 area 0
Device2(config-if-gigabitethernet0)#exit
Device2(config)#interface gigabitethernet1
Device2(config-if-gigabitethernet1)#ipv6 router ospf 100 area 0
Device2(config-if-gigabitethernet1)#exit
```

#Configure Device3.

```
Device3#configure terminal
Device3(config)#ipv6 router ospf 100
Device3(config-ospf6)#router-id 3.3.3.3
Device3(config-ospf6)#exit
Device3(config)#interface gigabitethernet0
Device3(config-if-gigabitethernet0)#ipv6 router ospf 100 area 0
```

```
Device3(config-if-gigabitethernet0)#exit
Device3(config)#interface gigabitethernet1
Device3(config-if-gigabitethernet1)#ipv6 router ospf 100 area 0
Device3(config-if-gigabitethernet1)#exit
Device3(config)#interface gigabitethernet2
Device3(config-if-gigabitethernet2)#ipv6 router ospf 100 area 0
Device3(config-if-gigabitethernet2)#exit
```

Step 3: Configure OSPFv3 to coordinate with BFD.

#Configure Device1.

```
Device1(config)#interface gigabitethernet0
Device1(config-if-gigabitethernet0)#ipv6 ospf bfd
Device1(config-if-gigabitethernet0)#exit
```

#Configure Device3.

```
Device3(config)#interface gigabitethernet0
Device3(config-if-gigabitethernet0)#ipv6 ospf bfd
Device3(config-if-gigabitethernet0)#exit
```

Step 4: Check the result.

#View the OSPFv3 neighbor information and route table of Device1.

```
Device1#show ipv6 ospf neighbor 3.3.3.3
OSPFv3 Process (100)

 Neighbor 3.3.3.3,interface address fe80::2212:1ff:fe01:104
   In the area 0.0.0.0 via interface gigabitethernet2, BFD enabled
   DR is 3.3.3.3 BDR is 1.1.1.1
   Neighbor priority is 1, State is Full, 6 state changes
   Options is 0x13 (-|R|-|-|E|V6)
   Dead timer due in 00:00:37
   Neighbor is up for 00:01:31
   Database Summary List 0
   Link State Request List 0
   Link State Retransmission List 0
   Thread Inactivity Timer on
   Thread Database Description Retransmission off, 0 times
   Thread Link State Request Retransmission off, 0 times
   Thread Link State Update Retransmission off, 0 times
```

```
Device1#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management

L   ::1/128 [0/0]
    via ::, 01:15:27, lo0
C   1001:1::/64 [0/0]
    via ::, 01:15:27, gigabitethernet2
L   1001:1::1/128 [0/0]
    via ::, 01:15:27, lo0
O   1001:2::/64 [110/2]
    via fe80::2212:1ff:fe01:104, 00:02:40, gigabitethernet0
C   2001:1::/64 [0/0]
    via ::, 01:15:27, gigabitethernet0
L   2001:1::1/128 [0/0]
    via ::, 01:15:27, lo0
C   2001:2::/64 [0/0]
    via ::, 01:15:27, gigabitethernet1
L   2001:2::1/128 [0/0]
    via ::, 01:15:27, lo0
O   2001:3::/64 [110/2]
    via fe80::201:7aff:fe5e:6d45, 00:02:40, gigabitethernet1
             [110/2]
    via fe80::2212:1ff:fe01:104, 00:02:40, gigabitethernet0
```

According to the OSPFv3 neighbor information, BFD has been enabled, and route 201.0.0.0/24 first selects the line between Device1 and Device3 for communication.

#View the BFD session of Device1.

```
Device1#show bfd session ipv6 detail
Total ipv6 session number: 1
OurAddr                       NeighAddr                      State      Holddown
Interface
fe80::201:7aff:fe61:7a25         fe80::2212:1ff:fe01:104            UP          5000
gigabitethernet0
Type:ipv6 direct
Local State:UP  Remote State:UP  Up for: 0h:0m:4s  Number of times UP:1
Local Discriminator:5  Remote Discriminator:95
```

```
Send Interval:1000ms  Detection time:5000ms(1000ms*5)
Local Diag:0  Demand mode:0  Poll bit:0
MinTxInt:1000  MinRxInt:1000  Multiplier:5
Remote MinTxInt:1000  Remote MinRxInt:1000  Remote Multiplier:5
Registered protocols:OSPFv3
```

OSPFv3 has successfully coordinated with BFD, and the session has been normally set up.

#If the line between Device1 and Device3 becomes faulty, BFD quickly detects the fault and informs OSPFv3 of the fault, and then, OSPFv3 switches the route to Device2 for communication. View the route table of Device1.

```
Device1#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management

L   ::1/128 [0/0]
    via ::, 01:16:10, lo0
C   1001:1::/64 [0/0]
    via ::, 01:16:10, gigabitethernet2
L   1001:1::1/128 [0/0]
    via ::, 01:16:10, lo0
O   1001:2::/64 [110/3]
    via fe80::201:7aff:fe5e:6d45, 00:00:07, gigabitethernet1
C   2001:1::/64 [0/0]
    via ::, 01:16:10, gigabitethernet0
L   2001:1::1/128 [0/0]
    via ::, 01:16:10, lo0
C   2001:2::/64 [0/0]
    via ::, 01:16:10, gigabitethernet1
L   2001:2::1/128 [0/0]
    via ::, 01:16:10, lo0
O   2001:3::/64 [110/2]
    via fe80::201:7aff:fe5e:6d45, 00:03:22, gigabitethernet1
```

The action of Device3 is similar to that of Device1.

7.3.4. Configure OSPFv3 Static Fast Re-routing

Network Requirements

- All devices are configured with the OSPFv3 protocol.
- Enable static fast rerouting between Device1 and Device3. When the line fails, the service can be quickly switched to Device2 for communication.

Network Topology

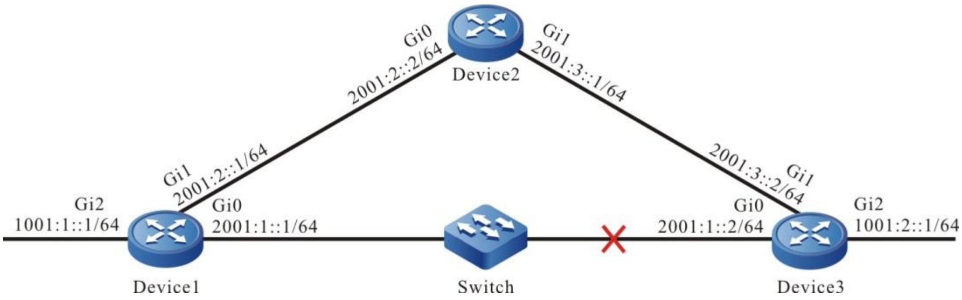

Figure 7-4 Networking of configuring OSPFv3 static fast re-routing

Configuration Steps

Step 1: Configure the IPv6 address of the interface (omitted).

Step 2: Configure the OSPFv3 process.

#Configure Device1.

```
Device1#configure terminal
Device1(config)#ipv6 router ospf 100
Device1(config-ospf6)#router-id 1.1.1.1
Device1(config-ospf6)#exit
Device1(config)#interface gigabitethernet0
Device1(config-if-gigabitethernet0)#ipv6 router ospf 100 area 0
Device1(config-if-gigabitethernet0)#exit
Device1(config)#interface gigabitethernet1
Device1(config-if-gigabitethernet1)#ipv6 router ospf 100 area 0
Device1(config-if-gigabitethernet1)#exit
```

#Configure Device2.

```
Device2#configure terminal
Device2(config)#ipv6 router ospf 100
Device2(config-ospf6)#router-id 2.2.2.2
Device2(config-ospf6)#exit
Device2(config)#interface gigabitethernet0
Device2(config-if-gigabitethernet0)#ipv6 router ospf 100 area 0
Device2(config-if-gigabitethernet0)#exit
Device2(config)#interface gigabitethernet1
Device2(config-if-gigabitethernet1)#ipv6 router ospf 100 area 0
Device2(config-if-gigabitethernet1)#exit
```

#Configure Device3.

```
Device3#configure terminal
Device3(config)#ipv6 router ospf 100
```

```
Device3(config-ospf6)#router-id 3.3.3.3
Device3(config-ospf6)#exit
Device3(config)#interface gigabitethernet0
Device3(config-if-gigabitethernet0)#ipv6 router ospf 100 area 0
Device3(config-if-gigabitethernet0)#exit
Device3(config)#interface gigabitethernet1
Device3(config-if-gigabitethernet1)#ipv6 router ospf 100 area 0
Device3(config-if-gigabitethernet1)#exit
Device3(config)#interface gigabitethernet2
Device3(config-if-gigabitethernet2)#ipv6 router ospf 100 area 0
Device3(config-if-gigabitethernet2)#exit
```

Step 3: On the interface gigabitethernet0 of Device3, configure the echo function of ipv6 bfd.

```
Device3(config)#interface gigabitethernet0
Device3(config-if-gigabitethernet0)# ipv6 bfd echo
Device3(config-if-gigabitethernet0)#exit
```

Step 4: Configure the routing policy.

#Configure Device1, configure route-map, and call ACL to match only 1001:2::1/64, while other network segments will be filtered out. The routing application matching the match rule backs up the next hop interface gigabitethernet1, and the next hop address 2001:2:: 2.

```
Device1(config)#ipv6 access-list extended 7001
Device1(config-v6-list)#permit ipv6 1001:2::1/64 any
Device1(config-v6-list)#exit
Device1(config)#route-map ipv6frr_ospf
Device1(config-route-map)#match ipv6 address 7001
Device1(config-route-map)#set ipv6 fast-reroute backup-interface gigabitethernet1
backup-nexthop 2001:2::2
Device1(config-route-map)#exit
```

Step 5: Configure the static fast re-routing.

```
Device1(config)#ipv6 router ospf 100
Device1(config-ospf6)#fast-reroute route-map ipv6frr_ospf
Device1(config-ospf6)#exit
```

Step 6: Check the result.

#View the OSPFv3 route table of Device1.

```
Device1#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
```

```
    O - OSPF, OE-OSPF External, M - Management
L  ::1/128 [0/0]
   via ::, 00:31:05, lo0
C  1001:1::/64 [0/0]
   via ::, 00:26:12, gigabitethernet2
L  1001:1::1/128 [0/0]
   via ::, 00:26:12, gigabitethernet2
O  1001:2::/64 [110/2]
   via fe80::201:7aff:fe92:e6b6, 00:22:11, gigabitethernet0
C  2001:1::/64 [0/0]
   via ::, 00:26:51, gigabitethernet0
L  2001:1::1/128 [0/0]
   via ::, 00:26:51, gigabitethernet0
C  2001:2::/64 [0/0]
   via ::, 00:25:30, gigabitethernet1
L  2001:2::1/128 [0/0]
   via ::, 00:25:30, gigabitethernet1
O  2001:3::/64 [110/2]
   via fe80::201:7aff:fe92:e6b6, 00:20:53, gigabitethernet0
   via fe80::ced8:1fff:fe10:7aae, 00:21:06, gigabitethernet1
```

It can be seen from the routing table that route 1001:2::/64 preferably communicates with the line between Device1 and Device3.

#View the IPv6 FRR route table of Device1.

```
Device1#show ipv6 frr route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
    U - Per-user Static route
    O - OSPF, OE-OSPF External, M - Management
O  1001:2::/64 [110/4294967295]
   via 2001:2::2, 00:03:16, gigabitethernet1
```

You can see that the next hop of the frr route 1001:2::/64 is 2001:2::2, and the outgoing interface is gigabitethernet1.

#View the BFD session information of Device1.

```
Device1#show bfd session ipv6 detail
Total ipv6 session number: 1
OurAddr                    NeighAddr          LD/RD         State     Holddown
Interface
fe80::201:7aff:fe61:7a25       fe80::2212:1ff:fe01:104    1062/1026      UP       5000
gigabitethernet0
```

```
Type:ipv6 direct  Mode:echo
Local Discriminator:65  Remote Discriminator:65
Local State:UP  Remote State:UP  Up for: 0h:9m:11s  Number of times UP:1
Send Interval:100ms  Detection time:500ms(100ms*5)
Local Diag:0  Demand mode:0  Poll bit:0
Registered protocols:FIB_MGR
Agent session info:
 Sender:slot 0  Recver:slot 0
```

You can see that FIB_MGR is associated with BFD successfully, the session is established normally, and the session mode is echo.

#After the line between Device1 and Device3 fails, it can quickly detect and switch to Device2 for communication. Check the routing table of Device1. In the routing table, the outgoing interface to the destination network 1001:2::/64 has been switched to the backup interface gigabitethernet1.

```
Device1#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management
L   ::1/128 [0/0]
    via ::, 03:03:45, lo0
C   1001:1::/64 [0/0]
    via ::, 02:58:52, gigabitethernet2
L   1001:1::1/128 [0/0]
    via ::, 02:58:52, gigabitethernet2
O   1001:2::/64 [110/3]
    via fe80::ced8:1fff:fe10:7aae, 00:00:11, gigabitethernet1
C   2001:1::/64 [0/0]
    via ::, 02:59:31, gigabitethernet0
L   2001:1::1/128 [0/0]
    via ::, 02:59:31, gigabitethernet0
C   2001:2::/64 [0/0]
    via ::, 02:58:10, gigabitethernet1
L   2001:2::1/128 [0/0]
    via ::, 02:58:10, gigabitethernet1
O   2001:3::/64 [110/2]
    via fe80::ced8:1fff:fe10:7aae, 02:53:45, gigabitethernet1
```

7.3.5. Configure OSPFv3 Dynamic Fast Re-routing

Network Requirements

- All devices are configured with the OSPFv3 protocol.

- Enable dynamic fast rerouting between Device1 and Device3. When the line fails, the service can be quickly switched to Device2 for communication.

Network Topology

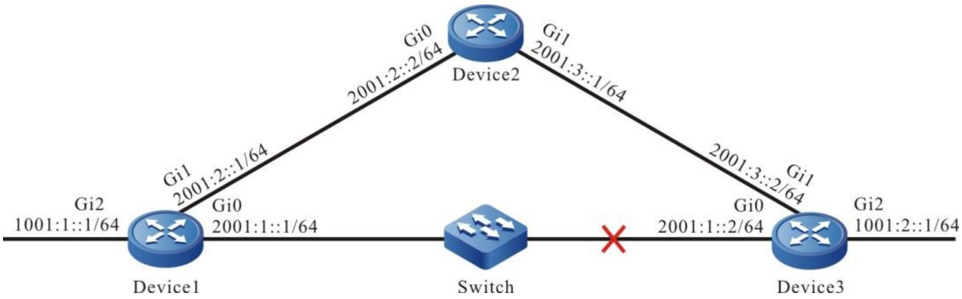

Figure 7-5 Networking of configuring OSPFv3 dynamic fast re-routing

Configuration Steps

Step 1: Configure the IPv6 address of the interface (omitted).

Step 2: Configure the OSPFv3 process and configure the interface network type as point-to-point.

#Configure Device1.

```
Device1#configure terminal
Device1(config)#ipv6 router ospf 100
Device1(config-ospf6)#router-id 1.1.1.1
Device1(config-ospf6)#exit
Device1(config)#interface gigabitethernet0
Device1(config-if-gigabitethernet0)#ipv6 router ospf 100 area 0
Device1(config-if-gigabitethernet0)# ipv6 ospf network point-to-point
Device1(config-if-gigabitethernet0)#exit
Device1(config)#interface gigabitethernet1
Device1(config-if-gigabitethernet1)#ipv6 router ospf 100 area 0
Device1(config-if-gigabitethernet1)# ipv6 ospf network point-to-point
Device1(config-if-gigabitethernet1)#exit
```

#Configure Device2.

```
Device2#configure terminal
Device2(config)#ipv6 router ospf 100
Device2(config-ospf6)#router-id 2.2.2.2
Device2(config-ospf6)#exit
Device2(config)#interface gigabitethernet0
Device2(config-if-gigabitethernet0)#ipv6 router ospf 100 area 0
Device2(config-if-gigabitethernet0)# ipv6 ospf network point-to-point
Device2(config-if-gigabitethernet0)#exit
Device2(config)#interface gigabitethernet1
```

```
Device2(config-if-gigabitethernet1)#ipv6 router ospf 100 area 0
Device2(config-if-gigabitethernet1)# ipv6 ospf network point-to-point
Device2(config-if-gigabitethernet1)#exit
```

#Configure Device3.

```
Device3#configure terminal
Device3(config)#ipv6 router ospf 100
Device3(config-ospf6)#router-id 3.3.3.3
Device3(config-ospf6)#exit
Device3(config)#interface gigabitethernet0
Device3(config-if-gigabitethernet0)#ipv6 router ospf 100 area 0
Device3(config-if-gigabitethernet0)# ipv6 ospf network point-to-point
Device3(config-if-gigabitethernet0)#exit
Device3(config)#interface gigabitethernet1
Device3(config-if-gigabitethernet1)#ipv6 router ospf 100 area 0
Device3(config-if-gigabitethernet1)# ipv6 ospf network point-to-point
Device3(config-if-gigabitethernet1)#exit
Device3(config)#interface gigabitethernet2
Device3(config-if-gigabitethernet2)#ipv6 router ospf 100 area 0
Device3(config-if-gigabitethernet2)# ipv6 ospf network point-to-point
Device3(config-if-gigabitethernet2)#exit
```

Step 3: On the interface gigabitethernet0 of Device3, configure the echo function of ipv6 bfd.

```
Device3(config)#interface gigabitethernet0
Device3(config-if-gigabitethernet0)# ipv6 bfd echo
Device3(config-if-gigabitethernet0)#exit
```

Step 4: Configure the routing policy.

#Configure Device1, configure route-map, and call ACL to only match 1001:2::1/64, while the other segments are filtered.

```
Device1(config)#ipv6 access-list extended 7001
Device1(config-v6-list)#permit ipv6 1001:2::1/64 any
Device1(config-v6-list)#exit
Device1(config)#route-map ipv6frr_ospf
Device1(config-route-map)#match ipv6 address 7001
Device1(config-route-map)#exit
```

Step 5: Configure the dynamic fast re-routing.

```
Device1(config)#ipv6 router ospf 100
Device1(config-ospf6)# fast-reroute loop-free-alternate route-map ipv6frr_ospf
```

```
Device1(config-ospf6)#exit
```

Step 6: Check the result.

#View the OSPFv3 route table of Device1.

```
Device1#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management
L   ::1/128 [0/0]
    via ::, 03:21:35, lo0
C   1001:1::/64 [0/0]
    via ::, 03:16:42, gigabitethernet2
L   1001:1::1/128 [0/0]
    via ::, 03:16:42, gigabitethernet2
O   1001:2::/64 [110/2]
    via fe80::201:7aff:fe92:e6b6, 00:01:35, gigabitethernet0
C   2001:1::/64 [0/0]
    via ::, 03:17:21, gigabitethernet0
L   2001:1::1/128 [0/0]
    via ::, 03:17:21, gigabitethernet0
C   2001:2::/64 [0/0]
    via ::, 03:16:00, gigabitethernet1
L   2001:2::1/128 [0/0]
    via ::, 03:16:00, gigabitethernet1
O   2001:3::/64 [110/2]
    via fe80::ced8:1fff:fe10:7aae, 00:07:50, gigabitethernet1
```

It can be seen from the routing table that route 1001:2::/64 preferably communicates with the line between Device1 and Device3.

#View the IPV6 FRR route table of Device1.

```
Device1#show ipv6 frr route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management
O   1001:2::/64 [110/4294967295]
    via fe80::ced8:1fff:fe10:7aae, 00:01:58, gigabitethernet1
```

You can see that the next hop of the frr route 1001:2::/64 is the linklocal address fe80:: ced8:1fff: fe10:7aae, and the outgoing interface is gigabitethernet1.

#View the BFD session information of Device1.

```
Device1#show bfd session ipv6 detail
Total ipv6 session number: 1
OurAddr                  NeighAddr            LD/RD         State     Holddown
Interface
fe80::201:7aff:fe61:7a25        fe80::2212:1ff:fe01:104    1062/1062       UP       5000
gigabitethernet0
Type:ipv6 direct  Mode:echo
Local Discriminator:66  Remote Discriminator:66
Local State:UP  Remote State:UP  Up for: 0h:4m:15s  Number of times UP:1
Send Interval:100ms  Detection time:500ms(100ms*5)
Local Diag:0  Demand mode:0  Poll bit:0
Registered protocols:FIB_MGR
Agent session info:
 Sender:slot 0  Recver:slot 0
```

You can see that FIB_MGR is associated with BFD successfully, the session is established normally, and the session mode is echo.

#After the line between Device1 and Device3 fails, it can quickly detect and switch to Device2 for communication. Check the routing table of Device1. In the routing table, the outgoing interface to the destination network 1001:2::/64 has been switched to the backup interface gigabitethernet1.

```
Device1#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management
L   ::1/128 [0/0]
    via ::, 03:03:45, lo0
C   1001:1::/64 [0/0]
    via ::, 02:58:52, gigabitethernet2
L   1001:1::1/128 [0/0]
    via ::, 02:58:52, gigabitethernet2
O   1001:2::/64 [110/3]
    via fe80::ced8:1fff:fe10:7aae, 00:00:11, gigabitethernet1
C   2001:1::/64 [0/0]
    via ::, 02:59:31, gigabitethernet0
L   2001:1::1/128 [0/0]
    via ::, 02:59:31, gigabitethernet0
C   2001:2::/64 [0/0]
    via ::, 02:58:10, gigabitethernet1
L   2001:2::1/128 [0/0]
    via ::, 02:58:10, gigabitethernet1
```

```
O   2001:3::/64 [110/2]
     via fe80::ced8:1fff:fe10:7aae, 02:53:45, gigabitethernet1
```

8. IS-IS

8.1. Overview

The IS-IS (Intermediate System to Intermediate System) is the IGP (Interior Gateway Protocol) based on the SPF algorithm. The basic design theory and algorithm for the IS-IS protocol are consistent with the OSPF. The IS-IS protocol is the routing protocol based on the link layer, which is irrelevant to the network layer (IPv4, IPv6, and OSI). It is not restricted by the network layer, and therefore it has good extensibility.

The IS-IS protocol can support the routing of multi-protocol stacks, including IPv4, IPv6, and OSI. The IS-IS protocol is initially applied to the OSI protocol stack (ISO10589) and then extended to the routing of IPv4 protocol stack (RFC1195) and IPv6 protocol stack (RFC5308).and, it can be further extended to support the CSPF calculation of the MPLS-TE (RFC3784).

The IS-IS protocol is characterized with good capability (inconsistent extended functions between devices can be compatible perfectly), large network capacity, able to support multi-protocol stacks, able to upgrade smoothly, unlikely to be faulty compared with the OSPF. Therefore, the IS-IS protocol applies to the large-size core backbone network. This section describes how to configure the IS-IS dynamic routing protocol on the device for network interconnection.

8.2. IS-IS Function Configuration

Table 8-1 IS-IS function list

Configuration Task	
Configure the IS-IS basic function	Enable the IS-IS protocol
	Configure the IS-IS VRF attribute
Configure the IS-IS layer attribute	Configure the IS-IS layer attribute
Configure the IS-IS route generation	Configure the IS-IS default route
	Configure the IS-IS routing redistribution
Configure the IS-IS routing control	Configure the IS-IS metric style
	Configure the IS-IS interface metric value
	Configure the IS-IS administrative distance
	Configure the IS-IS route summary

Configuration Task	
Configure the IS-IS routing control	Configure the maximum number of load-balanced routes for the IS-IS
	Configure the IS-IS inter-layer route leakage
	Configure the IS-IS ATT-bit
Configure the IS-IS network optimization	Configure the IS-IS interface priority
	Configure the IS-IS passive interface
	Configure the IS-IS Hello packet parameter
	Configure the IS-IS LSP packet parameter
	Configure the IS-IS SNP packet parameter
	Configure the IS-IS SPF calculation interval
	Configure the maximum number of areas for the IS-IS
	Configure the IS-IS host name mapping
	Configure the IS-IS interface to be added to the mesh group
Configure the IS-IS network authentication	Configure the IS-IS neighboring authentication
	Configure the IS-IS route authentication
Configure the IS-IS to coordinate with the BFD	Configure the IS-IS to coordinate with the BFD

8.2.1. Configure IS-IS Basic Function

Configuration Condition

Before using the IS-IS protocol, first complete the following tasks:

- Configure the link layer protocol to ensure the normal communication at the link layer.
- Configure the network layer IP address of the interface to enable the neighboring nodes to be reachable at the network layer.

Enable IS-IS Protocol

Multiple IS-IS processes can operate at the same time in the system. Each process is identified by different process names.

Table 8-2 Enable the IS-IS protocol

Step	Command	Description
Enter the global configuration mode	**configure terminal**	-
Create the IS-IS process and enter the IS-IS configuration mode	**router isis** [*area-tag*]	Mandatory By default, the IS-IS process does not operate in the system. The process name is *area-tag*.
Configure the network entity title for the IS-IS	**net** *entry-title*	Mandatory By default, the network entity title is not configured for the IS-IS.
Return to the global configuration mode	**exit**	-
Enter the interface configuration mode	**interface** *interface-name*	-
Enable the IS-IS protocol on the interface	**ip router isis** [*area-tag*]	Mandatory By default, the IS-IS protocol is not enabled on the interface.

Note:

- The IS-IS protocol cannot operate without the network entity title.

Configure IS-IS VRF Attribute

Multiple IS-IS processes can exist in the same VRF, but only one IS-IS process with Level-2 attribute in the VRF.

Table 8-3 Configure the IS-IS VRF attribute

Step	Command	Description
Enter the global configuration mode	**configure terminal**	-
Enter the IS-IS configuration mode	**router isis** [*area-tag*]	-
Enter the VRF attribute for the IS-IS	**vrf** *vrf-name*	Optional By default, the IS-IS process locates at the global VRF.

8.2.2. Configure IS-IS Layer Attribute

Configuration Condition

Before configuring the IS-IS layer attribute, first complete the following tasks:

- Configure the IP address of the interface to enable the neighboring nodes to be reachable at the network layer.
- Enable the IS-IS protocol.

Configure IS-IS Layer Attribute

The IS-IS layer attribute is divided into the global layer attribute and the interface layer attribute. The global layer attribute is the IS-IS intermediate system, which is further classified into the following three types:

- Level-1 intermediate system: Only the link status database of Level-1 is available and only the routing in the Level-1 area can be advertized and learnt.
- Level-2 intermediate system: Only the link status database of Level-2 is available and only the routing in the Level-2 area can be advertized and learnt.
- Level-1-2 intermediate system: Both the link status database of Level-1 and Level-2 are available and both the routing in the Level-1 and Level-2 area can be advertized and learnt. The Level-1-2 intermediate system is the interconnection device in the Level-1 and Level-2 area.

The layer attribute of the IS-IS interface is classified into the following three types:

- Level-1 attribute interface: Only the Level-1 packet of the IS-IS protocol can be transmitted and received and only the neighbor of Level-1 can be established.
- Level-2 attribute interface: Only the Level-2 packet of the IS-IS protocol can be transmitted and received and only the neighbor of Level-2 can be established.
- Level-1-2 attribute interface: Both the Level-1 packet and Level-2 packet of the IS-IS protocol can be transmitted and received and both the neighbors of Level-1 and Level-2 can be established.

The IS-IS interface layer attribute depends on the IS-IS global layer attribute. The Level-1 intermediate system only has the interface of Level-1 attribute, the Level-2 intermediate system only has the interface of Level-2 attribute, and the Level-1-2 intermediate system can has interfaces of all attributes.

Table 8-4 Configure the IS-IS global layer attribute

Step	Command	Description
Enter the global configuration mode	**configure terminal**	-
Enter the IS-IS configuration mode	**router isis** [*area-tag*]	-
Configure the IS-IS global layer attribute	**is-type { level-1 | level-1-2 | level-2-only }**	Optional By default, the IS-IS global layer attribute is Level-1-2.

Table 8-5 Configure the IS-IS interface layer attribute

Step	Command	Description
Enter the global configuration mode	**configure terminal**	-
Enter the interface configuration mode	**interface** *interface-name*	-
Configure the interface layer attribute	**isis circuit-type [level-1 | level-1-2 | level-2]**	Optional By default, the interface layer attribute is consistent with the global layer attribute when the interface layer attribute is not specified.

8.2.3. Configure IS-IS Route Generation

Configuration Condition

Before configuring the IS-IS route generation, first complete the following tasks:

- Configure the IP address for the interface to enable the neighboring nodes to be reachable at the network layer.
- Enable the IS-IS protocol.

Configure IS-IS Default Route

The Level-2 area of the IS-IS protocol cannot generate the default route during operating. You can configure to add a default route with the destination IP address as 0.0.0.0/0 in the Level-2 LSP and release it. When other areas of the same level in the intermediate system receive the route information, a default route will be added in the route table.

Table 8-6 Configure the IS-IS default route

Step	Command	Description
Enter the global configuration mode	**configure terminal**	-
Enter the IS-IS configuration mode	**router isis** [*area-tag*]	-
Enter the IS-IS IPv4 address family configuration mode	**address-family ipv4 unicast**	-
Configure the IS-IS to release he default route	**default-information originate**	Mandatory By default, the default route is not released.

Configure IS-IS Routing Redistribution

The routing redistribution can be used to introduce the routing information of other routing protocols to the IS-IS protocol. This enables the interconnection between the autonomous system of the IS-IS protocol and the autonomous system of other routing protocols or the routing area. When the external routing is introduced, the routing introduction policy and the routing layer attribute after introduction are specified.

Table 8-7 Configure the IS-IS routing redistribution

Step	Command	Description
Enter the global configuration mode	**configure terminal**	-
Enter the IS-IS configuration mode	**router isis** [*area-tag*]	-
Enter the IS-IS IPv4 address family configuration mode	**address-family ipv4 unicast**	-

Step	Command	Description
Configure the IS-IS routing redistribution	**redistribute** *protocol* [*protocol-id*] [**level-1** / **level-1-2** / **level-2** / **metric** *metric-value* / **metric-type** { **external** | **internal** } / **route-map** *route-map-name* / **match** *route-sub-type*]	Mandatory By default, information of other routing protocols are not redistributed.

8.2.4. Configure S-IS Routing Control

Configuration Condition

Before configuring the IS-IS routing feature, first complete the following tasks:

- Configure the IP address of the interface to enable the neighboring nodes to be reachable at the network layer.
- Enable the IS-IS protocol.

Configure IS-IS Metric Style

Initially, the IS-IS only has the narrow metric style. When the narrow metric style is used, the maximum metric value is 63. With the expansion of the network scale, the metric style cannot satisfy the requirements. Therefore, the wide metric style emerges whose metric value can reach 16777214. Devices use different metric styles cannot advertise and learn the routing information from each other. To realize the transition between the two metric styles, the configuration method for the transition metric style is provided.

The wide metric style is recommended.

Table 8-8 Configure the IS-IS metric style

Step	Command	Description
Enter the global configuration mode	**configure terminal**	-
Enter the IS-IS configuration mode	**router isis** [*area-tag*]	-
Configure the interface metric style	**metric-style** {**narrow** | **narrow transition** | **transition** | **wide** | **wide transition**} [**level-1** | **level-1-2** | **level-2**]	Optional By default, the narrow metric style is used.

Configure IS-IS Interface Metric Value

When the IS-IS protocol is enabled on the interface, the IS-IS routing metric is the global metric value. The following command can be used to specify a metric value for each interface.

Table 8-9 Configure the interface metric value

Step	Command	Description
Enter the global configuration mode	**configure terminal**	-
Enter the IS-IS configuration mode	**router isis** [*area-tag*]	-
Configure the IS-IS global metric value	**metric** *metric-value* **[level-1 | level-2]**	Optional By default, the global metric value is 10.
Return to the global configuration mode	**exit**	-
Enter the interface configuration mode	**interface** *interface-name*	-
Configure the interface metric value	**isis ipv4 metric** {*metric-value* | **maximum** } [**level-1** | **level-2**]	Optional By default, the global metric value is used.

Configure IS-IS Administrative Distance

The system chooses the primary routing based on the administrative distance. The smaller the administrative distance is, the higher priority the routing has.

Table 8-10 Configure the IS-IS administrative distance

Step	Command	Description
Enter the global configuration mode	**configure terminal**	-
Enter the IS-IS configuration mode	**router isis** [*area-tag*]	-

Step	Command	Description
Enter the IS-IS IPv4 address family configuration mode	**address-family ipv4 unicast**	-
Configure the IS-IS routing administrative distance	**Distance {** *distance-value / route-map name***}**	Optional By default, the administrative distance is 115.

Configure IS-IS Route Summary

The route summary summarizes multiple pieces of routing information as a piece of routing information. After the route summary is configured for the IS-IS, the number of advertisements to the subnet reduces effectively and the link status database and route table size reduce. This effectively saves the memory and CPU resources. This configuration is generally applied to the Level-1-2 edge device, reducing the routing information of layer advertisement.

Table 8-11 Configure the IS-IS route summary

Step	Command	Description
Enter the global configuration mode	**configure terminal**	-
Enter the IS-IS configuration mode	**router isis** [*area-tag*]	-
Enter the IPv4 address family configuration mode	**address-family ipv4 unicast**	-
Configure the IS-IS route summary	**summary-prefix** *prefix-value* [**metric** *metric-value* **/ route-type** {**internal | external**} **/ metric-type {internal | external**} **/ tag** *tag-value* **/ not-advertise / level-1 / level-2 / level-1-2]**	Mandatory By default, the route summary is not performed.

Configure the maximum number of load-balanced routes for the IS-IS

There are multiple paths of the same cost to the same destination IP address. These ECMP (equal cost multipath routing) can improve the link utilization rate. The user can control the maximum number of the IS-IS ECMPs.

Table 8-12 Configure the maximum number of load-balanced routes for the IS-IS

Step	Command	Description
Enter the global configuration mode	**configure terminal**	-
Enter the IS-IS configuration mode	**router isis** [*area-tag*]	-
Enter the IPv4 address family configuration mode	**address-family ipv4 unicast**	-
Configure the maximum number of load-balanced routes for the IS-IS	**maximum-paths** *max-number*	Optional By default, the maximum number of paths for load balancing is 4.

Configure IS-IS Inter-layer Route Leakage

By default, the IS-IS only leak the Level-1 routing to the Level-2, but the Level-1 area cannot know the routing of the Level-2 area. The inter-layer route leakage can be configured to introduce the Level-2 routing to the Level-1 area. When configuring the inter-layer route leakage, the routing policy can be specified to only leak the route that matches the condition.

Table 8-13 Configure the IS-IS inter-layer route leakage

Step	Command	Description
Enter the global configuration mode	**configure terminal**	-
Enter the IS-IS configuration mode	**router isis** [*area-tag*]	-
Enter the IS-IS IPv4 address family configuration mode	**address-family ipv4 unicast**	-
Configure the route leakage between the IS-IS layer	**propagate** { **level-1 into level-2** | **level-2 into level-1** } [**distribute-list** *access-list-name* | **route-map** *route-map-name*]	Mandatory By default, the Level-1 leaks its route to the Level-2.

Configure IS-IS ATT-bit

In the Level-1-2 device, the ATT-bit is used to inform other nodes whether this node has the connection to other areas. If yes, the ATT-bit will be set to 1 automatically and other nodes will generate a default route to this node. This increases the service load of this node. To avoid this situation, the ATT-bit can be forcibly set to 0.

Table 8-14 Configure the IS-IS ATT-bit

Step	Command	Description
Enter the global configuration mode	**configure terminal**	-
Enter the IS-IS configuration mode	**router isis** [*area-tag*]	-
Enter the IS-IS IPv4 address family configuration mode	**address-family ipv4 unicast**	-
Configure the IS-IS ATT-bit	**set-attached-bit** { **on** | **off** }	Mandatory By default, the ATT-bit is set based on whether the node is connected to other areas.

8.2.5. Configure IS-IS Network Optimization

Configuration Condition

Before configuring the IS-IS adjustment and optimization, first complete the following tasks:

- Configure the IP address of the interface to enable the neighboring nodes to be reachable at the network layer.
- Enable the IS-IS protocol.

Configure IS-IS Interface Priority

The IS-IS chooses a node on the broadcast link as the DIS node. The DIS node sends the CSNP packet periodically to synchronize the link status database on the entire network. The Level-1 and Level-2 select the DIS node respectively. The interface with the highest priority is selected as the DIS node. The node with the large MAC address is selected as the DIS node for the nodes with the same priority.

Table 8-15 Configure the IS-IS interface priority

Step	Command	Description
Enter the global configuration mode	**configure terminal**	-
Enter the interface configuration mode	**interface** *interface-name*	-
Configure the IS-IS interface priority	**isis priority** *priority-value* [**level-1** | **level-2**]	Optional By default, the interface priority is 64.

Configure IS-IS Passive Interface

The passive interface does not receive and transmit the IS-IS protocol packet, but still releases the directly connected network routing information of this interface. The IS-IS can reduce the bandwidth and CPU handling time through configuring the passive interface. Based on this configuration, the IS-IS can be specified to only release the directly connected network routing information of the passive interfaces and not release the directly connected network routing information of the non-passive interfaces.

Table 8-16 Configure the IS-IS passive interface

Step	Command	Description
Enter the global configuration mode	**configure terminal**	-
Enter the IS-IS configuration mode	**router isis** [*area-tag*]	-
Configure the IS-IS passive interface	**passive-interface** *interface-name*	Mandatory By default, the IS-IS does not have the passive interface.
Configure the IS-IS only to release the routing information of the passive interface	**advertise-passive-only**	Optional By default, the directly connected network routing information of the interface enabled with the IS-IS protocol is released.

Configure IS-IS Hello Packet Parameter

1. Configure the Hello packet delivery interval.

The interface enabled with the IS-IS protocol will send the Hello packet to keep the neighboring relationship with the neighboring devices. The smaller delivery interval of the Hello packet is, the faster the network convergence is. However, more bandwidth will be occupied.

Table 8-17 Configure delivery interval for the Hello packet

Step	**Command**	**Description**
Enter the global configuration mode	**configure terminal**	-
Enter the interface configuration mode	**interface** *interface-name*	-
Configure the Hello packet delivery interval on the interface	**isis hello-interval** { *interval* | **minimal** } [**level-1** | **level-2**]	Optional By default, the delivery interval of the Hello packet is 10s.

1. Configure the number of invalid Hello packets.

The IS-IS calculates the neighbor relationship retention time based on the number of invalid Hello packets and informs the retention time to the neighboring device. If the neighboring device does not receive the Hello packet from this device during this period, the neighbor relationship is invalid and the routing calculation will be recalculated.

Table 8-18 Configure the number of invalid Hello packets

Step	**Command**	**Description**
Enter the global configuration mode	**configure terminal**	-
Enter the interface configuration mode	**interface** *interface-name*	-
Configure the number of invalid Hello packets on the interface	**isis hello-multiplier** *multiplier* [**level-1** | **level-2**]	Optional By default, the number of invalid Hello packets is 3.

1. Configure to cancel the Hello packet padding function.

If MTU values on the interface at both sides of the link are inconsistent, as a result, smaller packets can be transmitted but larger packets cannot be transmitted. To avoid such situation, the IS-IS adopts the padding Hello packet to the interface MTU value to make the neighbor relationship cannot be established. However, this method wastes the bandwidth. In actual network, there is no need to configure the padding Hello packet. Only the small Hello packets are transmitted.

Table 8-19 Configure to cancel the Hello packet padding function

Step	Command	Description
Enter the global configuration mode	**configure terminal**	-
Enter the interface configuration mode	**interface** *interface-name*	-
Cancel the Hello packet padding function	**no isis hello padding**	Mandatory By default, the Hello packet padding function is enabled.

Configure IS-IS LSP Packet Parameter

1. Configure the maximum survival time for the LSP packet.

Each LSP packet has a maximum survival time. When the survival time of the LSP packet reduces to 0, the LSP packet will be deleted from the link status database. The maximum survival time of the LSP packet must be larger than the LSP packet refresh interval.

Table 8-20 Configure the IS-IS LSP packet parameter

Step	Command	Description
Enter the global configuration mode	**configure terminal**	-
Enter the IS-IS configuration mode	**router isis** [*area-tag*]	-
Configure the maximum survival time for the LSP packet	**max-lsp-lifetime** *life-time*	Optional By default, the maximum survival time of the LSP is 1200s.

1. Configure the LSP packet refresh interval.

The IS-IS protocol advertises and learns the routing through interacting each LSP packets. The nodes save the received LSP packets in the link status database. Each LSP packet has a maximum survival time and each node needs to refresh its LSP packet periodically to prevent the LSP packet maximum survival time reducing to 0 and keep the LSP packet in the entire area synchronization. Reducing the LSP packet delivery interval can accelerate the network convergence speed, but will occupy more bandwidth.

Table 8-21 Configure the LSP packet update packet

Step	Command	Description
Enter the global configuration mode	**configure terminal**	-
Enter the IS-IS configuration mode	**router isis** [*area-tag*]	-
Configure the LSP packet refresh interval	**lsp-refresh-interval** *refresh-interval*	Optional By default, the packet refresh interval for the periodical packet delivery is 900s.

1. Configure the LSP packet generation interval.

Periodical refresh will generate new LSP packet. Besides, the interface status changes and network status changes will also trigger new LSP packet generation. To prevent frequently generated LSP packets occupying too much CPU resources, the user can configure the minimum LSP packet generation interval.

Table 8-22 Configure the LSP packet generation interval

Step	Command	Description
Enter the global configuration mode	**configure terminal**	-
Enter the IS-IS configuration mode	**router isis** [*area-tag*]	-
Configure the LSP packet generation interval	**lsp-gen-interval** [**level-1** | **level-2**] *max-interval* [*initial-interval* [*secondary-interval*]]	Optional By default, the LSP packet generation interval is 50 ms.

1. Configure the LSP packet delivery interval.

Every generated LSP packet will be delivered on the interface. To avoid frequently generated LSP packet will greatly occupy the interface bandwidth. Each interface is configured with the minimum delivery interval of the LSP packet.

Table 8-23 Configure the LSP packet delivery interval

Step	**Command**	**Description**
Enter the global configuration mode	**configure terminal**	-
Enter the interface configuration mode	**interface** *interface-name*	-
Configure the LSP packet delivery interval	**isis lsp-interval** *min-interval*	Optional By default, the delivery interval of the LSP packet is 33 ms.

1. Configure the LSP packet retransmission time.

On the point-to-point link, the IS-IS sends the LSP packet and the then requires the peer end to send the PSNP acknowledgement message. If the IS-IS does not receive the acknowledgement message, the IS-IS will send the LSP packet again. The time waiting the acknowledgement message is the LSP packet retransmission interval. The retransmission interval can be set as required by the user to avoid LSP packet retransmission when the acknowledgement message is not received due to large delay.

Table 8-24 Configure the LSP packet retransmission time

Step	**Command**	**Description**
Enter the global configuration mode	**configure terminal**	-
Enter the interface configuration mode	**interface** *interface-name*	-
Configure the LSP packet retransmission time	**isis retransmit-interval** *interval* [**level-1** | **level-2**]	Optional By default, the retransmission time is 5s.

1. Configure the LSP MTU value.

The IS-IS protocol packet cannot perform automatic fragmentation. In order not to affecting normal LSP packet spread, the maximum length of the LSP packet in a routing domain cannot exceed the minimum MTU value on the IS-IS interfaces of all devices. Therefore, when the interface MTU values are inconsistent on devices in the routing domain, it is recommended that the maximum length of the LSP packet is set uniformly.

Table 8-25 Configure the LSP MTU value

Step	Command	Description
Enter the global configuration mode	**configure terminal**	-
Enter the IS-IS configuration mode	**router isis** [*area-tag*]	-
Configure the LSP packet MTU value	**lsp-mtu** *mtu-size* [**level-1** | **level-2**]	Optional By default, the MTU value of the LSP packet is 1492 bytes.

Configure IS-IS SNP Packet Parameter

1. Configure the CSNP packet delivery interval.

The selected nodes on the broadcast link need to send the CSNP packet periodically to synchronize the link status database on the entire network. The CSNP packet delivery interval is adjusted based on the actual situation.

Table 8-26 Configure the CSNP packet delivery interval

Step	Command	Description
Enter the global configuration mode	**configure terminal**	-
Enter the interface configuration mode	**interface** *interface-name*	-
Configure the CSNP packet delivery interval	**isis csnp-interval** *interval* [**level-1** | **level-2**]	Optional By default, the CSNP packet delivery interval is 10s.

1. Configure the PSNP packet delivery interval

On the broadcast link, the PSNP packet synchronizes the link status database on the entire network. On the point-to-point link, the PSNP packet confirms the received LSP packet. To avoid a large number of PSNP packets being delivered over the interface. A minimum delivery interval is set for the PSNP packet and the user can change the interval dynamically. The PSNP packet delivery interval cannot be set to a too large value. If the packet delivery interval is set to a too large value, the link status database synchronization on the entire network will be affected for the broadcast link, and the LSP packet may be redelivered caused by not timely receiving the acknowledgment message for the point-to-point link.

Table 8-27 Configure the PSNP packet delivery interval

Step	Command	Description
Enter the global configuration mode	**configure terminal**	-
Enter the interface configuration mode	**interface** *interface-name*	-
Configure the PSNP packet delivery interval	**isis psnp-interval** *min-interval* [**level-1** | **level-2**]	Optional By default, the PSNP packet delivery interval is 2s.

Configure IS-IS SPF Calculation Interval

The IS-IS link status database changes will trigger the SPF routing calculation. Frequent SPF calculation will consume a mass of CPU resources and user can configure the SPF calculation interval.

Table 8-28 Configure the IS-IS SPF calculation interval

Step	Command	Description
Enter the global configuration mode	**configure terminal**	-
Enter the IS-IS configuration mode	**router isis** [*area-tag*]	-
Enter the IS-IS IPv4 address family configuration mode	**address-family ipv4 unicast**	-
Configure the IS-IS SPF calculation interval	**spf-interval** [**level-1** | **level-2**] *maximum-interval* [*min-initial-delay* [*min-second-delay*]]	Optional By default, the *maximum-interval* is 10s, *min-initial-delay* is 50ms, *min-second-delay* is 200ms.

Configure Maximum Number of Areas for IS-IS

Multiple area IP addresses can be configured in an IS-IS process. Multiple area IP addresses are mainly applied in the following two situations that multiple Level-1 areas are combined as a Level-1 area, or a Level-1 area is divided into multiple Level-1 areas.

Table 8-29 Configure the maximum number of areas for the IS-IS

Step	Command	Description
Enter the global configuration mode	**configure terminal**	-
Enter the IS-IS configuration mode	**router isis** [*area-tag*]	-
Configure the maximum number of areas for the IS-IS	**max-area-addresses** *max-number*	Optional By default, the maximum number of the area IP addresses is 3.

Note:

- This configuration must be consistent in the entire IS-IS Level-1 routing domain. Otherwise, the Level-1 neighbor cannot be established normally. The Level-2 neighbor is not affected.

Configure IS-IS Host Name Mapping

The IS-IS uniquely identifies a intermediate system using the system ID with a fixed length of 6 bytes. When viewing the system information such as the neighbor relationship and link status database, the system ID cannot enable the user to visually associate the system ID with the host name. The IS-IS supports the mapping between the system ID and the host name to enable the user to view the system information more visually and conveniently. The IS-IS host name mapping can be configured in the following two methods:

1. Configure the IS-IS static host name mapping.

The IS-IS static host name mapping is manually established by the user between the system ID and the host name for the remote device .

Table 8-30 Configure the IS-IS static host name mapping

Step	Command	Description
Enter the global configuration mode	**configure terminal**	-
Enter the IS-IS configuration mode	**router isis** [*area-tag*]	-
Configure the IS-IS static hostname mapping	**hostname static** *system-id host-name*	Mandatory

1. Configure the IS-IS dynamic host name mapping.

The static host name mapping requires the user to configure the system ID and host name mapping of other devices on each device in the network, which has a heavy workload. The dynamic host name mapping only configures the host name for each device, and other devices in the network can learn the host name of the device when the host name advertisement function is enabled.

Table 8-31 Configure the IS-IS dynamic host name mapping

Step	Command	Description
Enter the global configuration mode	**configure terminal**	-
Enter the IS-IS configuration mode	**router isis** [*area-tag*]	-
Configure the IS-IS dynamic host name mapping	**hostname dynamic** { *host-name* | **area-tag** | **recv-only** | **system-name** }	Mandatory By default, only the host names advertized by other devices are learnt.

Configure IS-IS Interface to Be Added to Mesh Group

When the IS-IS interface is not added to the mesh group, the LSP packet received from an interface will be sent out on all the other IS-IS interfaces. This results in great bandwidth waste in a full mesh connected network. To avoid this situation, several IS-IS interfaces can be added to a mesh group. When an interface receives the LSP packet, it only sends the LSP packet out to the interface that is not in the same mesh group with this interface.

Table 8-32 Configure the IS-IS interface to be added to the mesh group

Step	Command	Description
Enter the global configuration mode	**configure terminal**	-
Enter the interface configuration mode	**interface** *interface-name*	-
Configure the IS-IS interface to be added to the mesh group	**isis mesh-group** { *group-number* | **blocked** }	Mandatory By default, the IS-IS interface is not added to the mesh group.

Note:

- The **isis mesh-group blocked** command can be used to configure the interface as the obstructive interface. The obstructive interface will not send the LSP packet actively and only send the LSP packet when receives the LSP request.

8.2.6. Configure IS-IS Network Authentication

Configuration Condition

Before configuring the IS-IS network authentication, first complete the following tasks:

- Configure the IP address of the interface to enable the neighboring nodes to be reachable at the network layer.
- Enable the IS-IS protocol.

Configure IS-IS Neighboring Authentication

When the neighbor relationship authentication is enabled for the IS-IS, the authentication information will be added to the delivered Hello packet and the received Hello packet will be authenticated. If the authentication fails, the neighbor relationship will not be established. This can prevent the neighbor relationship being established with the unreliable devices.

Table 8-33 Configure the IS-IS neighboring authentication

Step	Command	Description
Enter the global configuration mode	**configure terminal**	-
Enter the interface configuration mode	**interface** *interface-name*	-
Configure the Hello packet authentication mode	**isis authentication mode** { **md5** | **sm3** | **text** } [**level-1** | **level-2**]	Mandatory By default, the authentication function is not enabled.
Configure the Hello packet authentication password	**isis authentication key** { **0** | **7** } *password* [**level-1** | **level-2**]	Either By default, the authentication password is not configured. The authentication password can be configured using the password chain. For details about the password chain configuration, refer to the password chain configuration chapter in the configuration manual.
	isis authentication key-chain *key-chain-name* [**level-1** | **level-2**]	

Configure IS-IS Route Authentication

When the routing information authentication is enabled for the IS-IS, the authentication information will be added to the LSP and SNP packets and the received LSP and SNP packets will be authenticated. If the authentication fails, the packet will be dropped directly. This can prevent the unreliable routing information spreading to the IS-IS network.

Table 8-34 Configure the IS-IS route authentication

Step	Command	Description
Enter the global configuration mode	**configure terminal**	-
Enter the IS-IS configuration mode	**router isis** [*area-tag*]	-
Configure the authentication mode of the routing information packet	**authentication mode** { **md5** | **sm3** | **text** } [**level-1** | **level-2**]	Mandatory By default, the authentication function is not enabled.
Configure the authentication password of the routing information packet	**authentication key** { **0** | **7** } *password* [**level-1** | **level-2**]	Either By default, the authentication password is not configured. The authentication password can be configured using the password chain. For details about the password chain configuration, refer to the password chain configuration chapter in the configuration manual.
	authentication key-chain *key-chain-name* [**level-1** | **level-2**]	

8.2.7. Configure IS-IS to coordinate with the BFD

Configuration Condition

Before configuring the IS-IS to coordinate with the BFD, first complete the following task:

- Configure the IP address of the interface to enable the neighboring nodes to be reachable at the network layer.
- Enable the IS-IS protocol.

Configure IS-IS to Coordinate with BFD

The IS-IS to coordinate with the BFD is configure to fast detect the link faults and enable the backup link for communication. The IS-IS to coordinate with the BFD can be configured in the following two methods: All interfaces enabled with the IS-IS protocol are coordinated with the BFD and the interface is specified to coordinate with the BFD.

For details about the BFD parameter information, refer to the BFD configuration manual.

Table 8-35 Configure the IS-IS to coordinate with the BFD

Step	Command	Description
Enter the global configuration mode	**configure terminal**	-
Enter the interface configuration mode	**interface** *interface-name*	-
Configure the interface to enable the BFD link detection function	**isis bfd**	Mandatory By default, the BFD link detection function is not enabled for the interface.
Return to the global configuration mode	**exit**	-
Enter the IS-IS configuration mode	**router isis** [*area-tag*]	-
Configure all IS-IS interfaces to enable the BFD link detection function	**bfd all-interfaces**	Optional

8.2.8. Configure IS-IS Fast Re-routing

Configuration Condition

Before configuring IS-IS fast re-routing, ensure that:

- Interface IP addresses have been configured so that neighbor nodes are reachable at the network layer.
- Enable the IS-IS protocol.

Configure IS-IS Fast Re-routing

In the IS-IS network, if the link or device fails, the packet passing the fault point will be dropped or generate the loop and the caused traffic interruption will not recover until the protocol re-converges, which often lasts for several seconds. To reduce the traffic interruption time, you can configure the IS-IS fast re-routing. Apply the route map to set the backup next hop for the matched route. Once the active link fails, the traffic passing the faulty link will switch to the standby link at once.

Table 8-36 Configure the IS-IS fast re-routing

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the IS-IS configuration mode.	**router isis** [*area-tag*]	-
Enter the IS-IS IPv4 address family configuration mode	**address-family ipv4 unicast**	-
Configure ISIS to enable the fast re-routing function	**fast-reroute route-map** *route-map-name*	Mandatory By default, do not enable the IS-IS fast re-routing function.
Configure IS-IS to enable the dynamic fast re-routing function	**fast-reroute loop-free-alternate [route-map** *route-map-name*]	Mandatory By default, do not enable the IS-IS dynamic fast re-routing function.
Configure IS-IS to enable the pic function	**pic**	Mandatory After enabling the pic function, enable the auto fast re-routing function. By default, do not enable the IS-IS pic function.

Note:

- The IS-IS fast reroute function is divided into static fast reroute and dynamic fast reroute.
- The static fast rerouting function needs to associate with the route-map, and set the next hop interface and address of the backup route in the route-map.
- At present, dynamic fast reroute only supports point-to-point network, that is, the network type of all outgoing interfaces of the device needs to be point-to-point. After configuring dynamic fast reroute, the device automatically calculates and sets the backup next hop interface and address. Dynamic fast rerouting can also be associated with route-map. Only the routes matching route map are set to back up the next hop interface and address.
- The various modes of enabling rerouting are mutually exclusive;

8.2.9. IS-IS Monitoring and Maintaining

Table 8-37 The IS-IS monitoring and maintaining

Command	Description
clear isis [**instance -null** | *area-tag*] **statistics** [*interface_name*]	Clear the statistics information of the IS-IS protocol operation
clear isis [**instance -null** | *area-tag*] **process**	Restart the IS-IS protocol process
show isis [**instance -null** | *area-tag*]	Display the IS-IS process information
show isis instance { **-null** | *area-tag* } **bfd-sessions**	Display the BFD session information of the IS-IS process
show isis [**instance** -null | *area-tag*] **database** [*lsp_id*] [**detail**] [**l1** / **l2**] [**level-1** / **level-2**] [**self**] [**verbose**]	Display the IS-IS link status database information
show isis interface [*interface-name*] [**detail**]	Display the information of the IS-IS protocol interface operation
show isis [**instance –null** | *area-tag*] **ipv4 reach-info**	Display the IS-IS IPv4 subnet reachable information
show isis [**instance –null** | *area-tag*] **ipv4 route**	Display the IS-IS IPv4 routing information
show isis [**instance –null** | *area-tag*] **mesh-groups**	Display the IS-IS mesh group
show isis [**instance –null** | *area-tag*] **neighbors** [*interface-name*] [**detail**]	Display the IS-IS neighbor information
show isis [**instance –null** | *area-tag*] **statistics** [*interface-name*]	Display the statistics information of the IS-IS protocol operation
show isis router	Display the IS-IS host name information

8.3. IS-IS Typical Configuration Example

8.3.1. Configure IS-IS Basic Function

Network Requirements

- Configure the IS-IS protocol to realize the network interconnection between devices.
- Device1 is the Level-1 router and Device2 is the Level-1-2 router. Device1 and Device2 are in the same area, Area 10. Device3 is the Level-2 router in Area 20. Device2 connects the two areas.

Network Topology

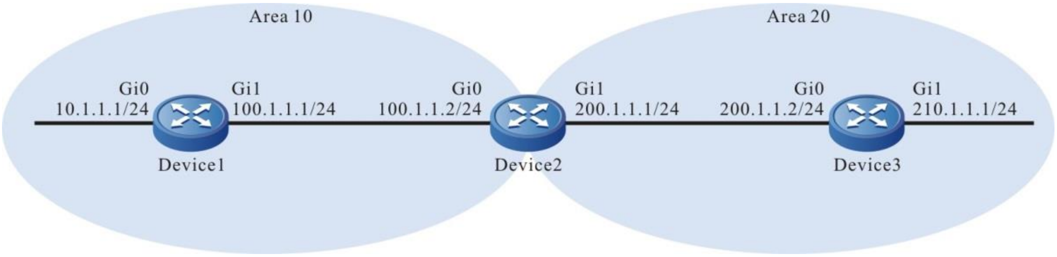

Figure 8–1 Networking of configuring the IS-IS basic functions

Configuration Steps

Step 1: Configure the IP address of the interfaces. (Omitted)

Step 2: Configure the IS-IS and enable the process on the interface.

#Configure the IS-IS process as 100, area number as 10, and type as Level-1 and enable the process on the interface on Device1.

```
Device1#configure terminal
Device1(config)#router isis 100
Device1(config-isis)#net 10.0000.0000.0001.00
Device1(config-isis)#is-type level-1
Device1(config-isis)#metric-style wide
Device1(config-isis)#exit
Device1(config)#interface gigabitethernet0
Device1(config-if-gigabitethernet0)#ip router isis 100
Device1(config-if-gigabitethernet0)#exit
Device1(config)#interface gigabitethernet1
Device1(config-if-gigabitethernet1)#ip router isis 100
Device1(config-if-gigabitethernet1)#exit
```

#Configure the IS-IS process as 100, area number as 10, and type as Level-1-2 and enable the process on the interface on Device2.

```
Device2#configure terminal
Device2(config)#router isis 100
Device2(config-isis)#net 10.0000.0000.0002.00
Device2(config-isis)#metric-style wide
```

```
Device2(config-isis)#exit
Device2(config)#interface gigabitethernet0
Device2(config-if-gigabitethernet0)#ip router isis 100
Device2(config-if-gigabitethernet0)#exit
Device2(config)#interface gigabitethernet1
Device2(config-if-gigabitethernet1)#ip router isis 100
Device2(config-if-gigabitethernet1)#exit
```

#Configure the IS-IS process as 100, area number as 20, and type as Level-2 and enable the process on the interface on Device3.

```
Device3#configure terminal
Device3(config)#router isis 100
Device3(config-isis)#net 20.0000.0000.0003.00
Device3(config-isis)#is-type level-2
Device3(config-isis)#metric-style wide
Device3(config-isis)#exit
Device3(config)#interface gigabitethernet0
Device3(config-if-gigabitethernet0)#ip router isis 100
Device3(config-if-gigabitethernet0)#exit
Device3(config)#interface gigabitethernet1
Device3(config-if-gigabitethernet1)#ip router isis 100
Device3(config-if-gigabitethernet1)#exit
```

Step 3: Check the result.

#View the IS-IS neighboring information of Device1.

```
Device1#show isis neighbors
IS-IS Instance 100 Neighbors (Counter 1):
Type   System ID      Interface        State Holdtime Level IETF-NSF Priority Circuit ID
L1-LAN  0000.0000.0002  gigabitethernet1      Up      29  sec     L1      capable   64
0000.0000.0001.01
```

#View the IS-IS neighboring information on Device2.

```
Device2#show isis neighbors
IS-IS Instance 100 Neighbors (Counter 2):
Type   System ID      Interface        State Holdtime Level IETF-NSF Priority Circuit ID
L2-LAN 0000.0000.0003 gigabitethernet1             Up     9 sec     L2     capable   64
0000.0000.0003.01
L1-LAN 0000.0000.0001 gigabitethernet0             Up     8 sec     L1     capable   64
0000.0000.0001.01
```

Device2 builds the IS-IS neighbor with Device1 and Device3, respectively.

#View the IS-IS neighboring information of Device3.

```
Device3#show isis neighbors
IS-IS Instance 100 Neighbors (Counter 1):
Type   System ID      Interface        State Holdtime Level IETF-NSF Priority Circuit ID
L2-LAN  0000.0000.0002  gigabitethernet0      Up      22  sec     L2      capable   64
0000.0000.0003.01
```

#View the routing information of Device1.

```
Device1#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

i   0.0.0.0/0 [115/10] via 100.1.1.2, 17:44:09, gigabitethernet1
C   10.1.1.0/24 is directly connected, 16:56:18, gigabitethernet0
L   10.1.1.1/32 is directly connected, 16:56:18, gigabitethernet0
C   100.1.1.0/24 is directly connected, 18:37:57, gigabitethernet1
L   100.1.1.1/32 is directly connected, 18:37:57, gigabitethernet1
C   127.0.0.0/8 is directly connected, 284:02:13, lo0
L   127.0.0.1/32 is directly connected, 284:02:13, lo0
i   200.1.1.0/24 [115/20] via 100.1.1.2, 17:44:09, gigabitethernet1

Device1#show isis ipv4 route
IS-IS Instance 100, VRF Kernel, IPv4 routes table (Counter 4):
L1 0.0.0.0/0, flags none, metric 10, from learned, installed
    via 100.1.1.2, gigabitethernet1, neighbor 0000.0000.0002
L1 10.1.1.0/24, flags none, metric 10, from network connected
    via 0.0.0.0, gigabitethernet0
L1 100.1.1.0/24, flags none, metric 10, from network connected
    via 0.0.0.0, gigabitethernet1
L1 200.1.1.0/24, flags none, metric 20, from learned, installed
    via 100.1.1.2, gigabitethernet1, neighbor 0000.0000.0002
```

A default routing is in the route table of Device1 and the next hop is Device2.

#View the routing information of Device2.

```
Device2#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external
```

```
i   10.1.1.0/24 [115/20] via 100.1.1.1, 16:58:26, gigabitethernet0
C   100.1.1.0/24 is directly connected, 18:39:58, gigabitethernet0
L   100.1.1.2/32 is directly connected, 18:39:58, gigabitethernet0
C   127.0.0.0/8 is directly connected, 20:16:34, lo0
L   127.0.0.1/32 is directly connected, 20:16:34, lo0
C   200.1.1.0/24 is directly connected, 18:39:37, gigabitethernet1
L   200.1.1.1/32 is directly connected, 18:39:37, gigabitethernet1
i   210.1.1.0/24 [115/20] via 200.1.1.2, 16:57:56, gigabitethernet1

Device2#show isis ipv4 route
IS-IS Instance 100, VRF Kernel, IPv4 routes table (Counter 4):
L1 10.1.1.0/24, flags none, metric 20, from learned, installed
    via 100.1.1.1, gigabitethernet0, neighbor 0000.0000.0001
L1 100.1.1.0/24, flags none, metric 10, from network connected
    via 0.0.0.0, gigabitethernet0
L1 200.1.1.0/24, flags none, metric 10, from network connected
    via 0.0.0.0, gigabitethernet1
L2 210.1.1.0/24, flags none, metric 20, from learned, installed
    via 200.1.1.2, gigabitethernet1, neighbor 0000.0000.0003
```

Device2 contains the Level-1 and Level-2 routing.

#View the routing information of Device3.

```
Device3#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

i   10.1.1.0/24 [115/30] via 200.1.1.1, 16:59:29, gigabitethernet0
i   100.1.1.0/24 [115/20] via 200.1.1.1, 17:47:29, gigabitethernet0
C   127.0.0.0/8 is directly connected, 945:29:12, lo0
L   127.0.0.1/32 is directly connected, 945:29:12, lo0
C   200.1.1.0/24 is directly connected, 18:40:27, gigabitethernet0
L   200.1.1.2/32 is directly connected, 18:40:27, gigabitethernet0
C   210.1.1.0/24 is directly connected, 16:59:04, gigabitethernet1
L   210.1.1.1/32 is directly connected, 16:59:04, gigabitethernet1

Device3#show isis ipv4 route
IS-IS Instance 100, VRF Kernel, IPv4 routes table (Counter 4):
```

```
L2 10.1.1.0/24, flags none, metric 30, from learned, installed
   via 200.1.1.1, gigabitethernet0, neighbor 0000.0000.0002
L2 100.1.1.0/24, flags none, metric 20, from learned, installed
   via 200.1.1.1, gigabitethernet0, neighbor 0000.0000.0002
L2 200.1.1.0/24, flags none, metric 10, from network connected
   via 0.0.0.0, gigabitethernet0
L2 210.1.1.0/24, flags none, metric 10, from network connected
   via 0.0.0.0, gigabitethernet1
```

Device3 learns the Level-1 routing and the Level-1 leaks the routing to Level-2 by default.

Note:

- The metric type is the narrow metric by default. The wide metric is recommended.
- The IS-IS entity attribute is Level-1-2 by default.

8.3.2. Configure IS-IS DIS Node Selection

Network Requirements

- Specify the device as the DIS node by changing the priority.
- Device1 and Device2 are the Level-1-2 devices, Device3 is the Level-1 device, and Device4 is the Level-2 device. Device1, Device2, Device3, and Device4 are in the same broadcast network and in the same area, Area 10.

Network Topology

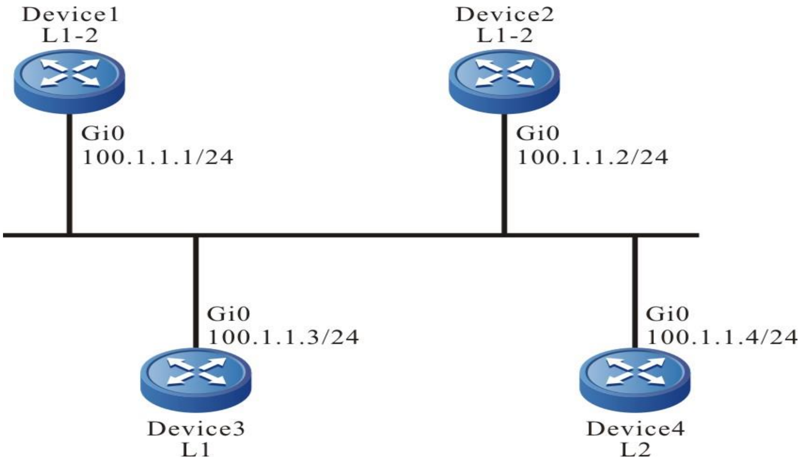

Figure 8–2 Networking of configuring the IS-IS DIS selection

Configuration Steps

Step 1: Configure the IP address of the interfaces. (Omitted)

Step 2: Configure the IS-IS and enable the process on the interface.

#Configure the IS-IS process as 100, area number as 10, and type as Level-1-2 and enable the process on the interface on Device1.

```
Device1#configure terminal
Device1(config)#router isis 100
Device1(config-isis)#net 10.0000.0000.0001.00
```

```
Device1(config-isis)#metric-style wide
Device1(config-isis)#exit
Device1(config)#interface gigabitethernet0
Device1(config-if-gigabitethernet0)#ip router isis 100
Device1(config-if-gigabitethernet0)#exit
```

#Configure the IS-IS process as 100, area number as 10, and type as Level-1-2 and enable the process on the interface on Device2.

```
Device2#configure terminal
Device2(config)#router isis 100
Device2(config-isis)#net 10.0000.0000.0002.00
Device2(config-isis)#metric-style wide
Device2(config-isis)#exit
Device2(config)#interface gigabitethernet0
Device2(config-if-gigabitethernet0)#ip router isis 100
Device2(config-if-gigabitethernet0)#exit
```

#Configure the IS-IS process as 100, area number as 10, and type as Level-1 and enable the process on the interface on Device3.

```
Device3#configure terminal
Device3(config)#router isis 100
Device3(config-isis)#net 10.0000.0000.0003.00
Device3(config-isis)#is-type level-1
Device3(config-isis)#metric-style wide
Device3(config-isis)#exit
Device3(config)#interface gigabitethernet0
Device3(config-if-gigabitethernet0)#ip router isis 100
Device3(config-if-gigabitethernet0)#exit
```

#Configure the IS-IS process as 100, area number as 20, and type as Level-2 and enable the process on the interface on Device4.

```
Device4#configure terminal
Device4(config)#router isis 100
Device4(config-isis)#net 20.0000.0000.0004.00
Device4(config-isis)#is-type level-2
Device4(config-isis)#metric-style wide
Device4(config-isis)#exit
Device4(config)#interface gigabitethernet0
Device4(config-if-gigabitethernet0)#ip router isis 100
Device4(config-if-gigabitethernet0)#exit
```

#View the IS-IS neighboring information of Device1.

```
Device1#show isis neighbors
IS-IS Instance 100 Neighbors (Counter 4):
Type   System ID      Interface        State Holdtime Level IETF-NSF Priority Circuit ID
L1-LAN 0000.0000.0002 gigabitethernet0            Up    23 sec   L1    capable  64
0000.0000.0003.01
L2-LAN 0000.0000.0002 gigabitethernet0            Up    23 sec   L2    capable  64
0000.0000.0004.01
L1-LAN 0000.0000.0003 gigabitethernet0            Up     8 sec   L1    capable  64
0000.0000.0003.01
L2-LAN 0000.0000.0004 gigabitethernet0            Up     8 sec   L2    capable  64
0000.0000.0004.01
```

The pseudo node of Level-1 is 0000.0000.0003.01 and Device3 is the DIS node of Level-1. The pseudo node of Level-2 is 0000.0000.0004.01 and Device1 is the DIS node of Level-2.

#Run the **show isis interface** command to view the MAC address of the interface. In the default priority, the DIS node is selected based on the principle that a larger MAC address of the physical interface has a higher priority.

Step 3: Modify the interface priority.

#Modify the interface priority of Device1.

```
Device1(config)#interface gigabitethernet0
Device1(config-if-gigabitethernet0)#isis priority 100
Device1(config-if-gigabitethernet0)#exit
```

Step 4: Check the result.

#View the IS-IS neighboring information of Device1.

```
Device1#show isis neighbors
IS-IS Instance 100 Neighbors (Counter 4):
Type   System ID      Interface        State Holdtime Level IETF-NSF Priority Circuit ID
L1-LAN 0000.0000.0002 gigabitethernet0            Up    24 sec   L1    capable  64
0000.0000.0001.01
L2-LAN 0000.0000.0002 gigabitethernet0            Up    23 sec   L2    capable  64
0000.0000.0001.01
L1-LAN 0000.0000.0003 gigabitethernet0            Up    20 sec   L1    capable  64
0000.0000.0001.01
L2-LAN 0000.0000.0004 gigabitethernet0            Up    24 sec   L2    capable  64
0000.0000.0001.01
```

The pseudo node of Level-1-2 is 0000.0000.0001.01 and Device1 is the DIS node of Levev-1-2.

Note:

- The IS-IS interface priority is 64 by default.

8.3.3. Configure IS-IS Inter-layer Route Leakage

Network Requirements

- Configure the inter-layer leakage on the Level-1-2 to leak the routing of Level-2 to Level-1.

Network Topology

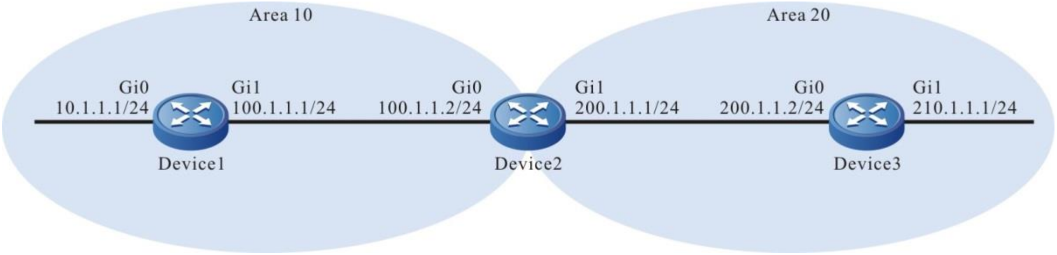

Figure 8–3 Networking of configuring the IS-IS inter-layer leakage

Configuration Steps

Step 1: Configure the IP address of the interfaces. (Omitted)

Step 2: Configure the IS-IS and enable the process on the interface.

#Configure the IS-IS process as 100, area number as 10, and type as Level-1 and enable the process on the interface on Device1.

```
Device1#configure terminal
Device1(config)#router isis 100
Device1(config-isis)#net 10.0000.0000.0001.00
Device1(config-isis)#is-type level-1
Device1(config-isis)#metric-style wide
Device1(config-isis)#exit
Device1(config)#interface gigabitethernet0
Device1(config-if-gigabitethernet0)#ip router isis 100
Device1(config-if-gigabitethernet0)#exit
Device1(config)#interface gigabitethernet1
Device1(config-if-gigabitethernet1)#ip router isis 100
Device1(config-if-gigabitethernet1)#exit
```

#Configure the IS-IS process as 100, area number as 10, and type as Level-1-2 and enable the process on the interface on Device2.

```
Device2#configure terminal
Device2(config)#router isis 100
Device2(config-isis)#net 10.0000.0000.0002.00
Device2(config-isis)#metric-style wide
Device2(config-isis)#exit
Device2(config)#interface gigabitethernet0
Device2(config-if-gigabitethernet0)#ip router isis 100
```

```
Device2(config-if-gigabitethernet0)#exit
Device2(config)#interface gigabitethernet1
Device2(config-if-gigabitethernet1)#ip router isis 100
Device2(config-if-gigabitethernet1)#exit
```

#Configure the IS-IS process as 100, area number as 20, and type as Level-2 and enable the process on the interface on Device3.

```
Device3#configure terminal
Device3(config)#router isis 100
Device3(config-isis)#net 20.0000.0000.0003.00
Device3(config-isis)#is-type level-2
Device3(config-isis)#metric-style wide
Device3(config-isis)#exit
Device3(config)#interface gigabitethernet0
Device3(config-if-gigabitethernet0)#ip router isis 100
Device3(config-if-gigabitethernet0)#exit
Device3(config)#interface gigabitethernet1
Device3(config-if-gigabitethernet1)#ip router isis 100
Device3(config-if-gigabitethernet1)#exit
```

#View the routing information of Device1.

```
Device1#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

i   0.0.0.0/0 [115/10] via 100.1.1.2, 17:44:09, gigabitethernet1
C   10.1.1.0/24 is directly connected, 16:56:18, gigabitethernet0
L   10.1.1.1/32 is directly connected, 16:56:18, gigabitethernet0
C   100.1.1.0/24 is directly connected, 18:37:57, gigabitethernet1
L   100.1.1.1/32 is directly connected, 18:37:57, gigabitethernet1
C   127.0.0.0/8 is directly connected, 284:02:13, lo0
L   127.0.0.1/32 is directly connected, 284:02:13, lo0
i   200.1.1.0/24 [115/20] via 100.1.1.2, 17:44:09, gigabitethernet1

Device1#show isis ipv4 route
IS-IS Instance 100, VRF Kernel, IPv4 routes table (Counter 4):
L1 0.0.0.0/0, flags none, metric 10, from learned, installed
     via 100.1.1.2, gigabitethernet1, neighbor 0000.0000.0002
```

```
L1 10.1.1.0/24, flags none, metric 10, from network connected
   via 0.0.0.0, gigabitethernet0
L1 100.1.1.0/24, flags none, metric 10, from network connected
   via 0.0.0.0, gigabitethernet1
L1 200.1.1.0/24, flags none, metric 20, from learned, installed
   via 100.1.1.2, gigabitethernet1, neighbor 0000.0000.0002

Device1#show isis database detail
IS-IS Instance 100 Level-1 Link State Database (Counter 3, LSP-MTU 1492):
LSPID                LSP Seq Num  LSP Checksum  LSP Holdtime   Length  ATT/P/OL
0000.0000.0001.00-00* 0x0000007E   0xD5DA        1067          71      0/0/0
  NLPID:       IPv4
  Area Address: 10
  IP Address:   100.1.1.1
  Metric:   10        IS-Extended 0000.0000.0001.01
  Metric:   10        IP-Extended 10.1.1.0/24
  Metric:   10        IP-Extended 100.1.1.0/24
0000.0000.0001.01-00* 0x00000073   0xAAAF        471           51      0/0/0
  Metric:   0         IS-Extended 0000.0000.0001.00
  Metric:   0         IS-Extended 0000.0000.0002.00
0000.0000.0002.00-00  0x00000081   0x5926        887           71      1/0/0
  NLPID:       IPv4
  Area Address: 10
  IP Address:   200.1.1.1
  Metric:   10        IS-Extended 0000.0000.0001.01
  Metric:   10        IP-Extended 100.1.1.0/24
  Metric:   10        IP-Extended 200.1.1.0/24
```

A default routing is in the route table and the next hop is Device2. No Level-2 routing advertised by Device3 is in the route table.

#View the routing information of Device2.

```
Device2#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

i   10.1.1.0/24 [115/20] via 100.1.1.1, 16:58:26, gigabitethernet0
C   100.1.1.0/24 is directly connected, 18:39:58, gigabitethernet0
```

```
L   100.1.1.2/24 is directly connected, 18:39:58, gigabitethernet0
C   127.0.0.0/8 is directly connected, 20:16:34, lo0
L   127.0.0.1/32 is directly connected, 20:16:34, lo0
C   200.1.1.0/24 is directly connected, 18:39:37, gigabitethernet1
L   200.1.1.1/32 is directly connected, 18:39:37, gigabitethernet1
i   210.1.1.0/24 [115/20] via 200.1.1.2, 16:57:56, gigabitethernet1

Device2#show isis ipv4 route
IS-IS Instance 100, VRF Kernel, IPv4 routes table (Counter 4):
L1 10.1.1.0/24, flags none, metric 20, from learned, installed
    via 100.1.1.1, gigabitethernet0, neighbor 0000.0000.0001
L1 100.1.1.0/24, flags none, metric 10, from network connected
    via 0.0.0.0, gigabitethernet0
L1 200.1.1.0/24, flags none, metric 10, from network connected
    via 0.0.0.0, gigabitethernet1
L2 210.1.1.0/24, flags none, metric 20, from learned, installed
    via 200.1.1.2, gigabitethernet1, neighbor 0000.0000.0003

Device2#show isis database detail
IS-IS Instance 100 Level-1 Link State Database (Counter 3, LSP-MTU 1492):
LSPID                LSP Seq Num  LSP Checksum  LSP Holdtime   Length  ATT/P/OL
0000.0000.0001.00-00  0x0000007E   0xD5DA       507          71      0/0/0
  NLPID:       IPv4
  Area Address: 10
  IP Address:   100.1.1.1
  Metric:   10         IS-Extended 0000.0000.0001.01
  Metric:   10         IP-Extended 10.1.1.0/24
  Metric:   10         IP-Extended 100.1.1.0/24
0000.0000.0001.01-00  0x00000074   0xA8B0       799          51      0/0/0
  Metric:   0          IS-Extended 0000.0000.0001.00
  Metric:   0          IS-Extended 0000.0000.0002.00
0000.0000.0002.00-00* 0x00000082   0x5727       1146         71      1/0/0
  NLPID:       IPv4
  Area Address: 10
  IP Address:   200.1.1.1
  Metric:   10         IS-Extended 0000.0000.0001.01
  Metric:   10         IP-Extended 100.1.1.0/24
  Metric:   10       IP-Extended 200.1.1.0/24

IS-IS Instance 100 Level-2 Link State Database (Counter 3, LSP-MTU 1492):
LSPID                LSP Seq Num  LSP Checksum  LSP Holdtime   Length  ATT/P/OL
0000.0000.0002.00-00* 0x00000081   0x84C0       1047          79      0/0/0
  NLPID:       IPv4
  Area Address: 10
  IP Address:   200.1.1.1
  Metric:   10       IS-Extended 0000.0000.0003.01
  Metric:   20        IP-Extended 10.1.1.0/24
  Metric:   10       IP-Extended 100.1.1.0/24
  Metric:   10       IP-Extended 200.1.1.0/24
0000.0000.0003.00-00  0x00000315   0x9DC7       543           71      0/0/0
  NLPID:       IPv4
  Area Address: 20
  IP Address:   210.1.1.1
  Metric:   10       IS-Extended 0000.0000.0003.01
  Metric:   10       IP-Extended 200.1.1.0/24
  Metric:   10       IP-Extended 210.1.1.0/24
0000.0000.0003.01-00  0x00000070   0xBF97       526           51      0/0/0
  Metric:   0        IS-Extended 0000.0000.0002.00
  Metric:   0        IS-Extended 0000.0000.0003.00
```

Device2 contains the Level-1 and Level-2 routing.

#View the routing information of Device3 and Device3 contains the Level-1 routing advertised by Device1.

```
Device3#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

i  10.1.1.0/24 [115/30] via 200.1.1.1, 16:59:29, gigabitethernet0
i  100.1.1.0/24 [115/20] via 200.1.1.1, 17:47:29, gigabitethernet0
C   127.0.0.0/8 is directly connected, 945:29:12, lo0
L   127.0.0.1/32 is directly connected, 945:29:12, lo0
C   200.1.1.0/24 is directly connected, 18:40:27, gigabitethernet0
L   200.1.1.2/32 is directly connected, 18:40:27, gigabitethernet0
C   210.1.1.0/24 is directly connected, 16:59:04, gigabitethernet1
```

```
L   210.1.1.1/32 is directly connected, 16:59:04, gigabitethernet1

Device3#show isis ipv4 route
IS-IS Instance 100, VRF Kernel, IPv4 routes table (Counter 4):
L2 10.1.1.0/24, flags none, metric 30, from learned, installed
    via 200.1.1.1, gigabitethernet0, neighbor 0000.0000.0002
L2 100.1.1.0/24, flags none, metric 20, from learned, installed
    via 200.1.1.1, gigabitethernet0, neighbor 0000.0000.0002
L2 200.1.1.0/24, flags none, metric 10, from network connected
    via 0.0.0.0, gigabitethernet0
L2 210.1.1.0/24, flags none, metric 10, from network connected
    via 0.0.0.0, gigabitethernet1

Device3#show isis database detail
IS-IS Instance 100 Level-2 Link State Database (Counter 3, LSP-MTU 1492):
LSPID                LSP Seq Num  LSP Checksum  LSP Holdtime   Length  ATT/P/OL
0000.0000.0002.00-00  0x00000081   0x84C0       880           79      0/0/0
  NLPID:       IPv4
  Area Address: 10
  IP Address:   200.1.1.1
  Metric:   10        IS-Extended 0000.0000.0003.01
  Metric:   20        IP-Extended 10.1.1.0/24
  Metric:   10        IP-Extended 100.1.1.0/24
  Metric:   10        IP-Extended 200.1.1.0/24
0000.0000.0003.00-00* 0x00000316   0x9BC8       1197          71      0/0/0
  NLPID:       IPv4
  Area Address: 20
  IP Address:   210.1.1.1
  Metric:   10        IS-Extended 0000.0000.0003.01
  Metric:   10        IP-Extended 200.1.1.0/24
  Metric:   10        IP-Extended 210.1.1.0/24
0000.0000.0003.01-00* 0x00000070   0xBF97       359           51      0/0/0
  Metric:   0         IS-Extended 0000.0000.0002.00
  Metric:   0         IS-Extended 0000.0000.0003.00
```

Step 3: Configure the inter-layer leakage.

#Configure the inter-layer leakage for Device2.

```
Device2(config)#router isis 100
Device2(config-isis)#address-family ipv4 unicast
Device2(config-isis-af)#propagate level-2 into level-1
Device2(config-isis-af)#exit
Device2(config-isis)#exit
```

Step 4: Check the result.

#View the routing information of Device1.

```
Device1#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
    U - Per-user Static route
    O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

i   0.0.0.0/0 [115/10] via 100.1.1.2, 17:44:09, gigabitethernet1
C   10.1.1.0/24 is directly connected, 16:56:18, gigabitethernet0
L   10.1.1.1/32 is directly connected, 16:56:18, gigabitethernet0
C   100.1.1.0/24 is directly connected, 18:37:57, gigabitethernet1
L   100.1.1.1/32 is directly connected, 18:37:57, gigabitethernet1
C   127.0.0.0/8 is directly connected, 284:02:13, lo0
L   127.0.0.1/32 is directly connected, 284:02:13, lo0
i   200.1.1.0/24 [115/20] via 100.1.1.2, 17:44:09, gigabitethernet1
i   210.1.1.0/24 [115/30] via 100.1.1.2, 00:00:01, gigabitethernet1

Device1#show isis ipv4 route
L1 0.0.0.0/0, flags none, metric 10, from learned, installed
   via 100.1.1.2, gigabitethernet1, neighbor 0000.0000.0002
L1 100.1.1.0/24, flags none, metric 10, from network connected
   via 0.0.0.0, gigabitethernet1
L1 200.1.1.0/24, flags none, metric 20, from learned, installed
   via 100.1.1.2, gigabitethernet1, neighbor 0000.0000.0002
L1 210.1.1.0/24, flags inter-area, metric 30, from learned, installed
   via 100.1.1.2, gigabitethernet1, neighbor 0000.0000.0002

Device1#show isis database detail
IS-IS Instance 100 Level-1 Link State Database (Counter 3, LSP-MTU 1492):
LSPID                 LSP Seq Num  LSP Checksum  LSP Holdtime   Length  ATT/P/OL
0000.0000.0001.00-00* 0x0000007F   0xD3DB        668            71      0/0/0
```

```
  NLPID:      IPv4
  Area Address: 10
  IP Address:   100.1.1.1
  Metric:   10       IS-Extended 0000.0000.0001.01
  Metric:   10       IP-Extended 10.1.1.0/24
  Metric:   10       IP-Extended 100.1.1.0/24
0000.0000.0001.01-00* 0x00000075   0xA6B1      995         51     0/0/0
  Metric:   0        IS-Extended 0000.0000.0001.00
  Metric:   0        IS-Extended 0000.0000.0002.00
0000.0000.0002.00-00  0x00000083   0x4DA6      984         79     1/0/0
  NLPID:      IPv4
  Area Address: 10
  IP Address:   200.1.1.1
  Metric:   10       IS-Extended 0000.0000.0001.01
  Metric:   10       IP-Extended 100.1.1.0/24
  Metric:   10       IP-Extended 200.1.1.0/24
  Metric:   20       IP-Extended ia 210.1.1.0/24
```

Besides the default routing, Device1 also learns the Level-2 routing advertised by Device3.

8.3.4. Configure IS-IS Routing Redistribution

Network Requirements

- Configure the redistribution to introduce the external routing to the IS-IS, enabling network interconnection between devices.
- Device1 and Device2 are the Level-2 routers. Configure the IS-IS and the Area 10. Configure the OSPF on Device2 and Device3. Redistribute the OSPF routing to the IS-IS through the configuration on Device2.

Network Topology

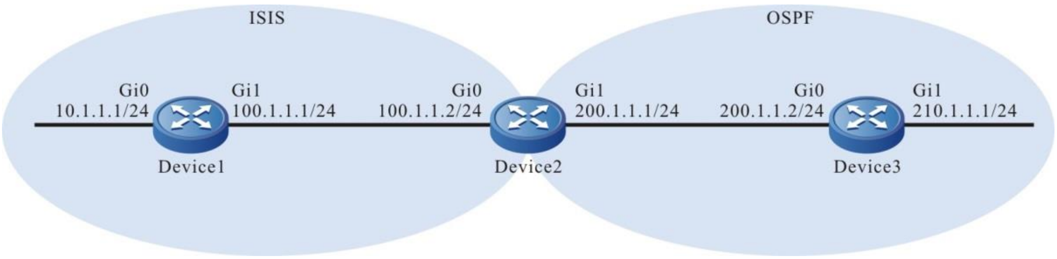

Figure 8 – 4 Networking of configuring the IS-IS routing redistribution

Configuration Steps

Step 1: Configure the IP address of the interfaces. (Omitted)

Step 2: Configure the IS-IS and enable the process on the interface.

#Configure the IS-IS process as 100, area number as 10, and type as Level-2 and enable the process on the interface on Device1.

```
Device1#configure terminal
Device1(config)#router isis 100
Device1(config-isis)#net 10.0000.0000.0001.00
Device1(config-isis)#is-type level-2
Device1(config-isis)#metric-style wide
Device1(config-isis)#exit
Device1(config)#interface gigabitethernet0
Device1(config-if-gigabitethernet0)#ip router isis 100
Device1(config-if-gigabitethernet0)#exit
Device1(config)#interface gigabitethernet1
Device1(config-if-gigabitethernet1)#ip router isis 100
Device1(config-if-gigabitethernet1)#exit
```

#Configure the IS-IS process as 100, area number as 10, and type as Level-1-2 and enable the process on the interface on Device2.

```
Device2#configure terminal
Device2(config)#router isis 100
Device2(config-isis)#net 10.0000.0000.0002.00
Device2(config-isis)#is-type level-2
Device2(config-isis)#metric-style wide
Device2(config-isis)#exit
Device2(config)#interface gigabitethernet0
Device2(config-if-gigabitethernet0)#ip router isis 100
Device2(config-if-gigabitethernet0)#exit
```

Step 3: Configure the OSPF.

#Configure Device2.

```
Device2(config)#router ospf 100
Device2(config-ospf)#network 200.1.1.0 0.0.0.255 area 0
Device2(config-ospf)#exit
```

#Configure Device3.

```
Device3#configure terminal
Device3(config)#router ospf 100
Device3(config-ospf)#network 210.1.1.0 0.0.0.255 area 0
Device3(config-ospf)#network 200.1.1.0 0.0.0.255 area 0
Device3(config-ospf)#exit
```

#View the route table of Device1. Device1 does not learn the OSPF routing redistributed by Device2.

```
Device1#show ip route
```

```
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

C  10.1.1.0/24 is directly connected, 00:58:39, gigabitethernet0
L  10.1.1.1/32 is directly connected, 00:58:39, gigabitethernet0
C  100.1.1.0/24 is directly connected, 06:55:35, gigabitethernet1
L  100.1.1.1/32 is directly connected, 06:55:35, gigabitethernet1
C  127.0.0.0/8 is directly connected, 603:06:22, lo0
L  127.0.0.1/32 is directly connected, 603:06:22, lo0

Device1#show isis ipv4 route
IS-IS Instance 100, VRF Kernel, IPv4 routes table (Counter 2):
L2 10.1.1.0/24, flags none, metric 10, from network connected
     via 0.0.0.0, gigabitethernet0
L2 100.1.1.0/24, flags none, metric 10, from network connected
     via 0.0.0.0, gigabitethernet1

Device1#show isis database detail
IS-IS Instance 100 Level-2 Link State Database (Counter 3, LSP-MTU 1492):
LSPID                LSP Seq Num  LSP Checksum  LSP Holdtime   Length  ATT/P/OL
0000.0000.0001.00-00* 0x00000046   0x489E       1123          71      0/0/0
  NLPID:       IPv4
  Area Address: 10
  IP Address:   100.1.1.1
  Metric:   10        IS-Extended 0000.0000.0001.01
  Metric:   10        IP-Extended 10.1.1.0/24
  Metric:   10        IP-Extended 100.1.1.0/24
0000.0000.0001.01-00* 0x00000045   0x097D       1103          51      0/0/0
  Metric:   0         IS-Extended 0000.0000.0001.00
  Metric:   0         IS-Extended 0000.0000.0002.00
0000.0000.0002.00-00  0x000000CB   0xEEA6       679           63      0/0/0
  NLPID:       IPv4
  Area Address: 10
  IP Address:   100.1.1.2
  Metric:   10        IS-Extended 0000.0000.0001.01
  Metric:   10        IP-Extended 100.1.1.0/24
```

#View the route table of Device2. Device2 learns the IS-IS and OSPF routing.

```
Device2#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

i  10.1.1.0/24 [115/20] via 100.1.1.1, 15:45:37, gigabitethernet0
C   100.1.1.0/24 is directly connected, 22:38:58, gigabitethernet0
L   100.1.1.2/32 is directly connected, 22:38:58, gigabitethernet0
C   127.0.0.0/8 is directly connected, 300:03:03, lo0
L   127.0.0.1/32 is directly connected, 300:03:03, lo0
C   200.1.1.0/24 is directly connected, 22:38:58, gigabitethernet1
L   200.1.1.1/32 is directly connected, 22:38:58, gigabitethernet1
O   210.1.1.1/32 [110/2] via 200.1.1.2, 15:43:35, gigabitethernet1

Device2#show isis ipv4 route
IS-IS Instance 100, VRF Kernel, IPv4 routes table (Counter 2):
L2 10.1.1.0/24, flags none, metric 20, from learned, installed
    via 100.1.1.1, gigabitethernet0, neighbor 0000.0000.0001
L2 100.1.1.0/24, flags none, metric 10, from network connected
    via 0.0.0.0, gigabitethernet0

Device2#show isis database detail
IS-IS Instance 100 Level-2 Link State Database (Counter 3, LSP-MTU 1492):
LSPID                LSP Seq Num  LSP Checksum  LSP Holdtime   Length  ATT/P/OL
0000.0000.0001.00-00  0x00000046   0x489E       911          71     0/0/0
  NLPID:       IPv4
  Area Address: 10
  IP Address:   100.1.1.1
  Metric:   10        IS-Extended 0000.0000.0001.01
  Metric:   10        IP-Extended 10.1.1.0/24
  Metric:   10        IP-Extended 100.1.1.0/24
0000.0000.0001.01-00  0x00000045   0x097D       892          51     0/0/0
  Metric:   0         IS-Extended 0000.0000.0001.00
  Metric:   0         IS-Extended 0000.0000.0002.00
0000.0000.0002.00-00* 0x000000CB   0xEEA6       467          63     0/0/0
  NLPID:       IPv4
```

```
    Area Address: 10
    IP Address:   100.1.1.2
    Metric:   10        IS-Extended 0000.0000.0001.01
    Metric:   10        IP-Extended 100.1.1.0/24
```

Step 4: Configure the IS-IS to redistribute the OSPF routing..

#Configure the OSPF routing to be redistributed to IS-IS Level-2 on Device2.

```
Device2(config)#router isis 100
Device2(config-isis)#address-family ipv4 unicast
Device2(config-isis-af)#redistribute ospf 100 level-2
Device2(config-isis-af)#exit
Device2(config-isis)#exit
```

Step 5: Check the result.

#View the routing information of Device1. Device1 leans the OSPF routing redistributed by Device2.

```
Device1#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

C   10.1.1.0/24 is directly connected, 16:47:30, gigabitethernet0
L   10.1.1.1/32 is directly connected, 16:47:30, gigabitethernet0
C   100.1.1.0/24 is directly connected, 22:44:27, gigabitethernet1
L   100.1.1.1/32 is directly connected, 22:44:27, gigabitethernet1
C   127.0.0.0/8 is directly connected, 618:55:13, lo0
L   127.0.0.1/32 is directly connected, 618:55:13, lo0
i   200.1.1.0/24 [115/10] via 100.1.1.2, 00:00:05, gigabitethernet1
i   210.1.1.1/32 [115/10] via 100.1.1.2, 00:00:05, gigabitethernet1

Device1#show isis ipv4 route
IS-IS Instance 100, VRF Kernel, IPv4 routes table (Counter 4):
L2 10.1.1.0/24, flags none, metric 10, from network connected
    via 0.0.0.0, gigabitethernet0
L2 100.1.1.0/24, flags none, metric 10, from network connected
    via 0.0.0.0, gigabitethernet1
L2 200.1.1.0/24, flags none, metric 10, from learned, installed
```

```
     via 100.1.1.2, gigabitethernet1, neighbor 0000.0000.0002
L2 210.1.1.1/32, flags none, metric 10, from learned, installed
     via 100.1.1.2, gigabitethernet1, neighbor 0000.0000.0002

Device1#show isis database detail
IS-IS Instance 100 Level-2 Link State Database (Counter 3, LSP-MTU 1492):
LSPID                LSP Seq Num  LSP Checksum  LSP Holdtime  Length  ATT/P/OL
0000.0000.0001.00-00* 0x00000046   0x489E       626          71      0/0/0
  NLPID:       IPv4
  Area Address: 10
  IP Address:   100.1.1.1
  Metric:   10       IS-Extended 0000.0000.0001.01
  Metric:   10       IP-Extended 10.1.1.0/24
  Metric:   10       IP-Extended 100.1.1.0/24
0000.0000.0001.01-00* 0x00000045   0x097D       606          51      0/0/0
  Metric:   0        IS-Extended 0000.0000.0001.00
  Metric:   0        IS-Extended 0000.0000.0002.00
0000.0000.0002.00-00  0x000000CD   0xC6E2       1184         80      0/0/0
  NLPID:       IPv4
  Area Address: 10
  IP Address:   100.1.1.2
  Metric:   10       IS-Extended 0000.0000.0001.01
  Metric:   10       IP-Extended 100.1.1.0/24
  Metric:   0        IP-Extended 200.1.1.0/24
  Metric:   0        IP-Extended 210.1.1.1/32
```

Device1 learns the redistributed OSPF routing.

8.3.5. Configure IS-IS Neighboring Authentication

Network Requirements

- Enable the authentication on the interface to enable the devices configured with the same password establishing the neighbor relationship.
- Device1 is the Level-1 router, Device2 is the Level-1-2 router, and Device1 and Device2 are in the same area, Area 10. Device3 is the Level-2 router in Area 20. Device2 connects the two areas.

Network Topology

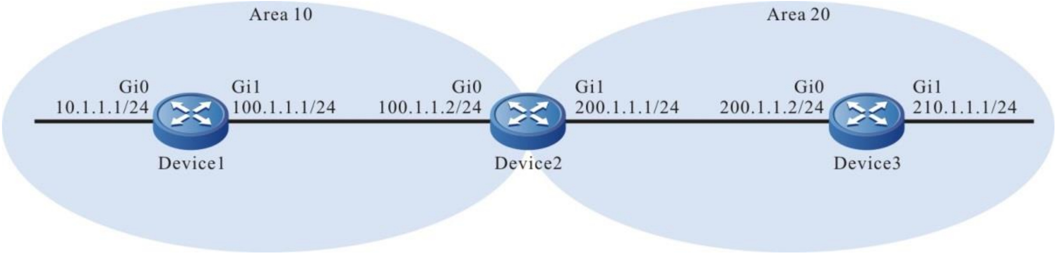

Figure 8-5 Networking of configuring the IS-IS neighbor authentication

Configuration Steps

Step 1: Configure the IP address of the interfaces. (Omitted)

Step 2: Configure the IS-IS and enable the process on the interface.

#Configure the IS-IS process as 100, area number as 10, and type as Level-1 and enable the process on the interface on Device1.

```
Device1#configure terminal
Device1(config)#router isis 100
Device1(config-isis)#net 10.0000.0000.0001.00
Device1(config-isis)#is-type level-1
Device1(config-isis)#metric-style wide
Device1(config-isis)#exit
Device1(config)#interface gigabitethernet0
Device1(config-if-gigabitethernet0)#ip router isis 100
Device1(config-if-gigabitethernet0)#exit
Device1(config)#interface gigabitethernet1
Device1(config-if-gigabitethernet1)#ip router isis 100
Device1(config-if-gigabitethernet1)#exit
```

#Configure the IS-IS process as 100, area number as 10, and type as Level-1-2 and enable the process on the interface on Device2.

```
Device2#configure terminal
Device2(config)#router isis 100
Device2(config-isis)#net 10.0000.0000.0002.00
Device2(config-isis)#metric-style wide
Device2(config-isis)#exit
Device2(config)#interface gigabitethernet0
Device2(config-if-gigabitethernet0)#ip router isis 100
Device2(config-if-gigabitethernet0)#exit
Device2(config)#interface gigabitethernet1
Device2(config-if-gigabitethernet1)#ip router isis 100
Device2(config-if-gigabitethernet1)#exit
```

#Configure the IS-IS process as 100, area number as 20, and type as Level-2 and enable the process on the interface on Device3.

```
Device3#configure terminal
Device3(config)#router isis 100
Device3(config-isis)#net 20.0000.0000.0003.00
Device3(config-isis)#is-type level-2
Device3(config-isis)#metric-style wide
Device3(config-isis)#exit
Device3(config)#interface gigabitethernet0
Device3(config-if-gigabitethernet0)#ip router isis 100
Device3(config-if-gigabitethernet0)#exit
Device3(config)#interface gigabitethernet1
Device3(config-if-gigabitethernet1)#ip router isis 100
Device3(config-if-gigabitethernet1)#exit
```

#View the IS-IS neighboring information of Device1.

```
Device1#show isis neighbors
IS-IS Instance 100 Neighbors (Counter 1):
Type   System ID      Interface       State Holdtime Level IETF-NSF Priority Circuit ID
L1-LAN  0000.0000.0002  gigabitethernet1     Up     29  sec    L1      capable   64
0000.0000.0001.01
```

#View the IS-IS neighboring information on Device2.

```
Device2#show isis neighbors
IS-IS Instance 100 Neighbors (Counter 2):
Type   System ID      Interface       State Holdtime Level IETF-NSF Priority Circuit ID
L2-LAN 0000.0000.0003 gigabitethernet1            Up    9 sec    L2    capable  64
0000.0000.0003.01
L2-LAN 0000.0000.0001 gigabitethernet0            Up    7 sec    L1    capable  64
0000.0000.0001.01
```

#View the IS-IS neighboring information of Device3.

```
Device3#show isis neighbors
IS-IS Instance 100 Neighbors (Counter 1):
Type   System ID      Interface       State Holdtime Level IETF-NSF Priority Circuit ID
L2-LAN  0000.0000.0002  gigabitethernet0     Up     24  sec    L2      capable   64
0000.0000.0003.01
```

Step 3: Configure the authentication.

#Configure the MD5 authentication and password admin on the interface of Device2.

```
Device2(config)#interface gigabitethernet0
Device2(config-if-gigabitethernet0)#isis authentication mode md5
```

```
Device2(config-if-gigabitethernet0)#isis authentication key 0 admin
Device2(config-if-gigabitethernet0)#exit
Device2(config)#interface gigabitethernet1
Device2(config-if-gigabitethernet1)#isis authentication mode md5
Device2(config-if-gigabitethernet1)#isis authentication key 0 admin
Device2(config-if-gigabitethernet1)#exit
```

#View the IS-IS neighbor of Device2.

```
Device2#show isis neighbors
IS-IS Instance 100 Neighbors (Counter 0):
Type   System ID      Interface        State Holdtime Level IETF-NSF Priority Circuit ID
At this time, Device1 and Device3 are not configured with the authentication. Device2
does not establish the IS-IS neighbor.
```

#Configure the MD5 authentication and password admin on the gigabitethernet1 interface of Device1.

```
Device1(config)#interface gigabitethernet1
Device1(config-if-gigabitethernet1)#isis authentication mode md5
Device1(config-if-gigabitethernet1)#isis authentication key 0 admin
Device1(config-if-gigabitethernet1)#exit
```

#Configure the MD5 authentication and password admin on the gigabitethernet0 interface of Device3.

```
Device3(config)#interface gigabitethernet0
Device3(config-if-gigabitethernet0)#isis authentication mode md5
Device3(config-if-gigabitethernet0)#isis authentication key 0 admin
Device3(config-if-gigabitethernet0)#exit
```

Step 4: Check the result.

#View the IS-IS neighboring information of Device1.

```
Device1#show isis neighbors
IS-IS Instance 100 Neighbors (Counter 1):
Type   System ID      Interface        State Holdtime Level IETF-NSF Priority Circuit ID
L1-LAN  0000.0000.0002  gigabitethernet1      Up      29  sec     L1      capable    64
0000.0000.0001.01
```

It can be observed that the IS-IS neighbor is successfully established between Device1 and Device2. It indicates that the authentication succeeds.

#View the IS-IS neighboring information on Device2.

```
Device2#show isis neighbors
IS-IS Instance 100 Neighbors (Counter 2):
Type   System ID      Interface        State Holdtime Level IETF-NSF Priority Circuit ID
```

```
L2-LAN 0000.0000.0003 gigabitethernet1           Up    9 sec    L2    capable  64
0000.0000.0003.01
L2-LAN 0000.0000.0001 gigabitethernet0           Up    7 sec    L1    capable  64
0000.0000.0001.01
```

It can be observed that the IS-IS neighbor is successfully established between Device2 and Device1/Device3. It indicates that the authentication succeeds.

#View the IS-IS neighboring information of Device3.

```
Device3#show isis neighbors
IS-IS Instance 100 Neighbors (Counter 1):
Type   System ID      Interface       State Holdtime Level IETF-NSF Priority Circuit ID
L2-LAN 0000.0000.0002 gigabitethernet0      Up     24 sec    L2     capable   64
0000.0000.0003.01
```

It can be observed that the IS-IS neighbor is successfully established between Device3 and Device2. It indicates that the authentication succeeds.

#View the routing information of Device2. Device2 can normally receives the routing advertised by Device1 and Device3.

```
Device2#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

i   10.1.1.0/24 [115/20] via 100.1.1.1, 16:58:26, gigabitethernet0
C   100.1.1.0/24 is directly connected, 18:39:58, gigabitethernet0
L   100.1.1.2/32 is directly connected, 18:39:58, gigabitethernet0
C   127.0.0.0/8 is directly connected, 20:16:34, lo0
L   127.0.0.1/32 is directly connected, 20:16:34, lo0
C   200.1.1.0/24 is directly connected, 18:39:37, gigabitethernet1
L   200.1.1.1/32 is directly connected, 18:39:37, gigabitethernet1
i   210.1.1.0/24 [115/20] via 200.1.1.2, 16:57:56, gigabitethernet1

Device2#show isis ipv4 route
IS-IS Instance 100, VRF Kernel, IPv4 routes table (Counter 4):
L1 10.1.1.0/24, flags none, metric 20, from learned, installed
    via 100.1.1.1, gigabitethernet0, neighbor 0000.0000.0001
L1 100.1.1.0/24, flags none, metric 10, from network connected
    via 0.0.0.0, gigabitethernet0
L1 200.1.1.0/24, flags none, metric 10, from network connected
    via 0.0.0.0, gigabitethernet1
L2 210.1.1.0/24, flags none, metric 20, from learned, installed
```

```
via 200.1.1.2, gigabitethernet1, neighbor 0000.0000.0003
```

8.3.6. Configure IS-IS to Coordinate with BFD

Network Requirements

- Configure the BFD coordination between devices. When the main line is faulty, services can be quickly switched to the backup line.
- Device1, Device2, and Device3 are the Level-2 routers in the same area, Area 10. Configure the BFD on Device1 and Device3 to initiate a session. When the line between Device1 and Device3 disconnects, Device1 can perform switching quickly and learn the 10.1.1.1/24 routing from Device2.

Network Topology

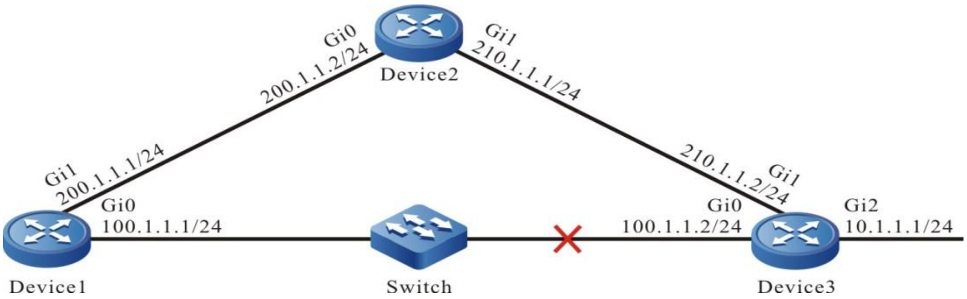

Figure 8–6 Networking of configuring the IS-IS coordinating with the BFD

Configuration Steps

Step 1: Configure the IP address of the interfaces. (Omitted)

Step 2: Configure the IS-IS and enable the process on the interface.

#Configure the IS-IS process as 100, area number as 10, and type as Level-2 and enable the process on the interface on Device1.

```
Device1#configure terminal
Device1(config)#router isis 100
Device1(config-isis)#net 10.0000.0000.0001.00
Device1(config-isis)#is-type level-2
Device1(config-isis)#metric-style wide
Device1(config-isis)#exit
Device1(config)#interface gigabitethernet0
Device1(config-if-gigabitethernet0)#ip router isis 100
Device1(config-if-gigabitethernet0)#exit
Device1(config)#interface gigabitethernet1
Device1(config-if-gigabitethernet1)#ip router isis 100
Device1(config-if-gigabitethernet1)#exit
```

#Configure the IS-IS process as 100, area number as 10, and type as Level-2 and enable the process on the interface on Device2.

```
Device2#configure terminal
Device2(config)#router isis 100
```

```
Device2(config-isis)#net 10.0000.0000.0002.00
Device2(config-isis)#is-type level-2
Device2(config-isis)#metric-style wide
Device2(config-isis)#exit
Device2(config)#interface gigabitethernet0
Device2(config-if-gigabitethernet0)#ip router isis 100
Device2(config-if-gigabitethernet0)#exit
Device2(config)#interface gigabitethernet1
Device2(config-if-gigabitethernet1)#ip router isis 100
Device2(config-if-gigabitethernet1)#exit
```

#Configure the IS-IS process as 100, area number as 10, and type as Level-2 and enable the process on the interface on Device3.

```
Device3#configure terminal
Device3(config)#router isis 100
Device3(config-isis)#net 10.0000.0000.0003.00
Device3(config-isis)#is-type level-2
Device3(config-isis)#metric-style wide
Device3(config-isis)#exit
Device3(config)#interface gigabitethernet0
Device3(config-if-gigabitethernet0)#ip router isis 100
Device3(config-if-gigabitethernet0)#exit
Device3(config)#interface gigabitethernet1
Device3(config-if-gigabitethernet1)#ip router isis 100
Device3(config-if-gigabitethernet1)#exit
Device3(config)#interface gigabitethernet2
Device3(config-if-gigabitethernet2)#ip router isis 100
Device3(config-if-gigabitethernet2)#exit
```

#View the routing information of Device1. Device1 preferentially chooses the routing 10.1.1.0/24 advertised by Device3.

```
Device1#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

i  10.1.1.0/24 [115/20] via 100.1.1.2, 00:00:15, gigabitethernet0
C   100.1.1.0/24 is directly connected, 00:09:15, gigabitethernet0
L   100.1.1.1/32 is directly connected, 00:09:15, gigabitethernet0
C   127.0.0.0/8 is directly connected, 253:58:17, lo0
```

```
L   127.0.0.1/32 is directly connected, 253:58:17, lo0
C   200.1.1.0/24 is directly connected, 00:11:29, gigabitethernet1
L   200.1.1.1/32 is directly connected, 00:11:29, gigabitethernet1
i   210.1.1.0/24 [115/20] via 100.1.1.2, 00:00:15, gigabitethernet0
               [115/20] via 200.1.1.2, 00:00:15, gigabitethernet1
Device1#show isis ipv4 route
IS-IS Instance 100, VRF Kernel, IPv4 routes table (Counter 4):
L2 10.1.1.0/24, flags none, metric 20, from learned, installed
    via 100.1.1.2, gigabitethernet0, neighbor 0000.0000.0003
L2 100.1.1.0/24, flags none, metric 10, from network connected
    via 0.0.0.0, gigabitethernet0
L2 200.1.1.0/24, flags none, metric 10, from network connected
    via 0.0.0.0, gigabitethernet1
L2 210.1.1.0/24, flags none, metric 20, from learned, installed
    via 100.1.1.2, gigabitethernet0, neighbor 0000.0000.0003
    via 200.1.1.2, gigabitethernet1, neighbor 0000.0000.0002
```

Step 3: Configure the BFD.

#Enable the BFD on the interface of Device1.

```
Device1(config)#interface gigabitethernet0
Device1(config-if-gigabitethernet0)#isis bfd
Device1(config-if-gigabitethernet0)#exit
```

#Enable the BFD on the interface of Device3.

```
Device3(config)#interface gigabitethernet0
Device3(config-if-gigabitethernet0)#isis bfd
Device3(config-if-gigabitethernet0)#exit
```

#View the BFD information of Device1.

```
Device1#show bfd session
OurAddr             NeighAddr           LD/RD           State       Holddown
interface
100.1.1.2       100.1.1.1       1/1       UP       5000       gigabitethernet0
```

Step 4: Check the result.

#When the line between Device1 and Device3 is faulty, the BFD quickly detects the fault and informs the fault to the IS-IS. The ISIS switches the routing to Device2 for communication. View the route table of Device1.

```
Device1#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
```

```
          O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

i   10.1.1.0/24 [115/30] via 200.1.1.2, 00:00:14, gigabitethernet1
C   127.0.0.0/8 is directly connected, 112:55:25, lo0
L   127.0.0.1/32 is directly connected, 112:55:25, lo0
C   200.1.1.0/24 is directly connected, 101:20:08, gigabitethernet1
L   200.1.1.1/32 is directly connected, 101:20:08, gigabitethernet1

Device1#show isis ipv4 route
IS-IS Instance 100, VRF Kernel, IPv4 routes table (Counter 2):
L2 10.1.1.0/24, flags none, metric 30, from l earned, installed
    via 200.1.1.2, gigabitethernet1, neighbor 0000.0000.0003
L2 200.1.1.0/24, flags none, metric 20, from network connected
    via 0.0.0.0, gigabitethernet1
```

#It can be viewed that the data flow from Device1 to the 10.1.1.0/24 network segment uses the line with Device2 to forward.

8.3.7. Configure ISIS Fast Re-routing

Network Requirements

- All devices configure the ISIS protocol.
- Device1 learns the ISIS route 192.168.1.1/32 from Device2 and Device3 at the same time. Device1 first uses the line with Device3 to forward the packet. Similarly, Device3 learns the ISIS route 100.1.1.1/32 from Device1 and Device2 at the same time. Device3 first uses the line with Device1 to forward the packet.
- Device1 and Device3 enable the ISIS fast re-routing. After the line between Device1 and Device3 fails, the service can switch to Device2 for communication fast.

Network Topology

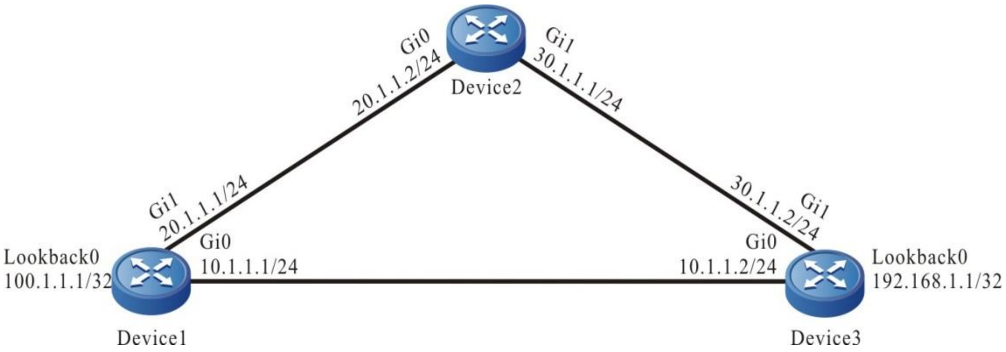

Figure 8–7 Configure the ISIS fast re-routing

Configuration Steps

Step 1: Configure the IP addresses of the interfaces. (Omitted)

Step 2: Configure ISIS and enable the process on the interface.

#Device1 configures the ISIS process 100, the area ID is 10, the type is Level-2, and enable the process on the interface.

```
Device1#configure terminal
Device1(config)#router isis 100
Device1(config-isis)#net 10.0000.0000.0001.00
Device1(config-isis)#is-type level-2
Device1(config-isis)#metric-style wide
Device1(config-isis)#exit
Device1(config)#interface gigabitethernet0
Device1(config-if-gigabitethernet0)#ip router isis 100
Device1(config-if-gigabitethernet0)#exit
Device1(config)#interface gigabitethernet1
Device1(config-if-gigabitethernet1)#ip router isis 100
Device1(config-if-gigabitethernet1)#exit
Device1(config)#interface loopback 0
Device1(config-if-loopback0)#ip router isis 100
Device1(config-if-loopback0)#exit
```

#Device2 configures the ISIS process 100, the area ID is 10, the type is Level-2, and enable the process on the interface.

```
Device2#configure terminal
Device2(config)#router isis 100
Device2(config-isis)#net 10.0000.0000.0002.00
Device2(config-isis)#is-type level-2
Device2(config-isis)#metric-style wide
Device2(config-isis)#exit
Device2(config)#interface gigabitethernet0
Device2(config-if-gigabitethernet0)#ip router isis 100
Device2(config-if-gigabitethernet0)#exit
Device2(config)#interface gigabitethernet1
Device2(config-if-gigabitethernet1)#ip router isis 100
Device2(config-if-gigabitethernet1)#exit
```

#Device3 configures the ISIS process 100, the area ID is 10, the type is Level-2, and enable the process on the interface.

```
Device3#configure terminal
Device3(config)#router isis 100
Device3(config-isis)#net 10.0000.0000.0003.00
Device3(config-isis)#is-type level-2
Device3(config-isis)#metric-style wide
```

```
Device3(config-isis)#exit
Device3(config)#interface gigabitethernet0
Device3(config-if-gigabitethernet0)#ip router isis 100
Device3(config-if-gigabitethernet0)#exit
Device3(config)#interface gigabitethernet1
Device3(config-if-gigabitethernet1)#ip router isis 100
Device3(config-if-gigabitethernet1)#exit
Device3(config)#interface loopback 0
Device3(config-if-loopback0)#ip router isis 100
Device3(config-if-loopback0)#exit
```

Step 3: Configure the route policy.

#Configure Device1: configure route-map to call the ACL only matching 192.168.1.1/32, while the other network is filtered. The route matching the match rule applies the backup next-hop interface gigabitethernet1 and the next-hop address is 20.1.1.2.

```
Device1(config)#ip access-list standard 1
Device1(config-std-nacl)#permit 192.168.1.1 0.0.0.0
Device1(config-std-nacl)#exit
Device1(config)#route-map ipfrr_isis
Device1(config-route-map)#match ip address 1
Device1(config-route-map)#set fast-reroute backup-interface gigabitethernet1
backup-nexthop 20.1.1.2
Device1(config-route-map)#exit
```

#Configure Device3: configure route-map to call the ACL only matching 100.1.1.1/32, while the other network is filtered. The route matching the match rule applies the backup next-hop interface gigabitethernet1 and the next-hop address is 30.1.1.1.

```
Device3(config)#ip access-list standard 1
Device3(config-std-nacl)#permit 100.1.1.1 0.0.0.0
Device3(config-std-nacl)#exit
Device3(config)#route-map ipfrr_isis
Device3(config-route-map)#match ip address 1
Device3(config-route-map)#set fast-reroute backup-interface gigabitethernet1
backup-nexthop 30.1.1.1
Device3(config-route-map)#exit
```

Step 4: Configure the fast re-routing.

#Configure Device1 to enable the ISIS fast re-routing.

```
Device1(config)#router isis 100
```

```
Device1(config-isis)#address-family ipv4 unicast
Device1(config-isis-af)#ipfrr route-map ipfrr_isis
Device1(config-isis-af)#exit-address-family
Device1(config-isis)#exit
```

#Configure Device3 to enable the ISIS fast re-routing.

```
Device3(config)#router isis 100
Device3(config-isis)#address-family ipv4 unicast
Device3(config-isis-af)#ipfrr route-map ipfrr_isis
Device3(config-isis-af)#exit-address-family
Device3(config-isis)#exit
```

Step 5: Check the result.

#View the route table of Device1.

```
Device1#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external
C   10.1.1.0/24 is directly connected, 00:02:39, gigabitethernet0
L   10.1.1.1/32 is directly connected, 00:02:39, gigabitethernet0
C   20.1.1.0/24 is directly connected, 06:19:51, gigabitethernet1
L   20.1.1.1/32 is directly connected, 06:19:51, gigabitethernet1
i   30.1.1.0/24 [115/20] via 20.1.1.2, 00:00:03, gigabitethernet1
               [115/20] via 10.1.1.2, 00:00:03, gigabitethernet0
C   127.0.0.0/8 is directly connected, 30:54:34, lo0
L   127.0.0.1/32 is directly connected, 30:54:34, lo0
LC  100.1.1.1/32 is directly connected, 02:54:43, loopback0
i   192.168.1.1/32 [115/20] via 10.1.1.2, 00:01:13, gigabitethernet0
```

#View the fast re-route table of Device1 and you can see that there is the route of the network 192.168.1.1/32 and the next-hop interface is gigabitethernet1.

```
Device1#show ip frr route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external
i   192.168.1.1/32 [115/0] via 20.1.1.2, 00:00:03, gigabitethernet1
```

#View the backup next-hop information of Device1 and the fast re-routing backup interface is gigabitethernet1.

```
Device1#show nexthop frr detail
```

```
Index                : 262
Type                 : FRR
Reference Count         : 1
Active Path             : master
Nexthop Address         : 10.1.1.2
Interface               : gigabitethernet0
Interface Vrf           : global
Channel ID              : 10
Link Header Length      : 18
Link Header             : 00017abc662b201201010101810000010800
Action                  : FORWORDING
Slot                    : 0
BK Nexthop Address      : 20.1.1.2
BK Interface            : gigabitethernet1
BK Interface Vrf        : global
BK Channel ID           : 11
BK Link Header Length   : 18
BK Link Header          : 00017a455449201201010102810000020800
BK Action               : FORWORDING
BK Slot                 : 0

Total 1 entries.
```

#After the line between Device1 and Device3 fails, the system can fast detect and switch to Device2 for communication. View the route table and fast re-route table of Device1. The egress interface to the destination network 192.168.1.1/32 in the route table is switched to the backup interface gigabitethernet1.

```
Device1#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

C   10.1.1.0/24 is directly connected, 00:02:39, gigabitethernet0
L   10.1.1.1/32 is directly connected, 00:02:39, gigabitethernet0
C   20.1.1.0/24 is directly connected, 06:19:51, gigabitethernet1
L   20.1.1.1/32 is directly connected, 06:19:51, gigabitethernet1
i   30.1.1.0/24 [115/20] via 20.1.1.2, 00:00:03, gigabitethernet1
               [115/20] via 10.1.1.2, 00:00:03, gigabitethernet0
C   127.0.0.0/8 is directly connected, 30:54:34, lo0
```

```
L   127.0.0.1/32 is directly connected, 30:54:34, lo0
LC  100.1.1.1/32 is directly connected, 02:54:43, loopback0
i   192.168.1.1/32 [115/30] via 20.1.1.2, 00:01:13, gigabitethernet1
```

The processing mode of Device3 is similar with Device1.

9. IPV6 IS-IS

9.1. Overview

IPv6 IS-IS routing protocol and IS-IS routing protocol have the same behaviors except for the IP address structure in the packet. Refer to the brief introduction of IS-IS routing protocol.

9.2. IPv6 IS-IS Function Configuration

Table 9-1 IPV6 IS-IS function list

Configuration Task	
Configure the IPV6 IS-IS basic function	Enable the IPV6 IS-IS protocol
	Configure the IPV6 IS-IS VRF attribute
Configure the IPV6 IS-IS layer attribute	Configure the IPV6 IS-IS layer attribute
Configure the IPV6 IS-IS route generation	Configure the IPV6 IS-IS default route
	Configure the IPV6 IS-IS routing redistribution
Configure the IPV6 IS-IS routing control	Configure the IPV6 IS-IS metric style
	Configure the IPV6 IS-IS interface metric value
	Configure the IPV6 IS-IS administrative distance
	Configure the IPV6 IS-IS route summary
	Configure the maximum number of load-balanced routes for the IPV6 IS-IS
	Configure the IPV6 IS-IS inter-layer route leakage
	Configure the IPV6 IS-IS ATT-bit

Configuration Task	
Configure the IPV6 IS-IS network optimization	Configure the IPV6 IS-IS interface priority
	Configure the IPV6 IS-IS passive interface
	Configure the IPV6 IS-IS Hello packet parameter
	Configure the IPV6 IS-IS LSP packet parameter
	Configure the IPV6 IS-IS SNP packet parameter
	Configure the IPV6 IS-IS SPF calculation interval
	Configure the maximum number of areas for the IPV6 IS-IS
	Configure the IPV6 IS-IS host name mapping
	Configure the IPV6 IS-IS interface to be added to the mesh group
Configure the IPV6 IS-IS network authentication	Configure the IPV6 IS-IS neighboring authentication
	Configure the IPV6 IS-IS route authentication
Configure the IPV6 IS-IS fast re-routing	Configure the IPV6 IS-IS fast re-routing

9.2.1. Configure IPV6 IS-IS Basic Function

Configuration Condition

Before using the IPV6 IS-IS protocol, first complete the following tasks:

- Configure the link layer protocol to ensure the normal communication at the link layer.
- Configure the network layer IP address of the interface to enable the neighboring nodes to be reachable at the network layer.

Enable IPV6 IS-IS Protocol

Multiple IPV6 IS-IS processes can operate at the same time in the system. Each process is identified by different process names.

Table 9-2 Enable the IPV6 IS-IS protocol

Step	**Command**	**Description**
Enter the global configuration mode	**configure terminal**	-
Create the IPV6 IS-IS process and enter the IPV6 IS-IS configuration mode	**router isis** [*area-tag*]	Mandatory By default, the IPV6 IS-IS process does not operate in the system. The process name is *area-tag*.
Configure the network entity title for the IPV6 IS-IS	**net** *entry-title*	Mandatory By default, the network entity title is not configured for the IPV6 IS-IS.
Return to the global configuration mode	**exit**	-
Enter the interface configuration mode	**interface** *interface-name*	-
Enable the IPV6 IS-IS protocol on the interface	**ipv6 router isis** [*area-tag*]	Mandatory By default, the IPV6 IS-IS protocol is not enabled on the interface.

Note:

- The IS-IS protocol cannot operate without the network entity title.

Configure IS-IS VRF Attribute

Table 9-3 Configure the IS-IS VRF attribute

Step	**Command**	**Description**
Enter the global configuration mode	**configure terminal**	-
Enter the IPV6 IS-IS configuration mode	**router isis** [*area-tag*]	-

Step	Command	Description
Enter the VRF attribute for the IS-IS	**vrf** *vrf-name*	Optional By default, the IS-IS process locates at the global VRF.

9.2.2. Configure IPV6 IS-IS Layer Attribute

Configuration Condition

Before configuring the IPV6 IS-IS layer attribute, first complete the following tasks:

- Configure the IPv6 address of the interface to enable the neighboring nodes to be reachable at the network layer.
- Enable the IPV6 IS-IS protocol.

Configure IS-IS Layer Attribute

The IS-IS layer attribute is divided into the global layer attribute and the interface layer attribute. The global layer attribute is the IS-IS intermediate system, which is further classified into the following three types:

- Level-1 intermediate system: Only the link status database of Level-1 is available and only the routing in the Level-1 area can be advertised and learnt.
- Level-2 intermediate system: Only the link status database of Level-2 is available and only the routing in the Level-2 area can be advertised and learnt.
- Level-1-2 intermediate system: Both the link status database of Level-1 and Level-2 are available and both the routing in the Level-1 and Level-2 area can be advertised and learnt. The Level-1-2 intermediate system is the interconnection device in the Level-1 and Level-2 area.

The layer attribute of the IS-IS interface is classified into the following three types:

- Level-1 attribute interface: Only the Level-1 packet of the IS-IS protocol can be transmitted and received and only the neighbor of Level-1 can be established.
- Level-2 attribute interface: Only the Level-2 packet of the IS-IS protocol can be transmitted and received and only the neighbor of Level-2 can be established.
- Level-1-2 attribute interface: Both the Level-1 packet and Level-2 packet of the IS-IS protocol can be transmitted and received and both the neighbors of Level-1 and Level-2 can be established.

The IS-IS interface layer attribute depends on the IS-IS global layer attribute. The Level-1 intermediate system only has the interface of Level-1 attribute, the Level-2 intermediate system only has the interface of Level-2 attribute, and the Level-1-2 intermediate system can has interfaces of all attributes.

Table 9-4 Configure the IPV6 IS-IS global layer attribute

Step	Command	Description
Enter the global configuration mode	**configure terminal**	-
Enter the IS-IS configuration mode	**router isis** [*area-tag*]	-
Configure the IS-IS global layer attribute	**is-type { level-1 | level-1-2 | level-2-only }**	Optional By default, the IS-IS global layer attribute is Level-1-2.

Table 9-5 Configure the IS-IS interface layer attribute

Step	Command	Description
Enter the global configuration mode	**configure terminal**	-
Enter the interface configuration mode	**interface** *interface-name*	-
Configure the interface layer attribute	**isis circuit-type [level-1 | level-1-2 | level-2-only]**	Optional By default, the interface layer attribute is consistent with the global layer attribute when the interface layer attribute is not specified.

9.2.3. Configure IPV6 IS-IS Route Generation

Configuration Condition

Before configuring the IPV6 IS-IS route generation, first complete the following tasks:

- Configure the IPv6 address for the interface to enable the neighboring nodes to be reachable at the network layer.
- Enable the IPV6 IS-IS protocol.

Configure IPV6 IS-IS Default Route

The Level-2 area of the IS-IS protocol cannot generate the default route during operating. You can configure to add a default route with the destination IP address as 0.0.0.0/0 in the Level-2

LSP and release it. When other areas of the same level in the intermediate system receive the route information, a default route will be added in the routing table.

Table 9-6 Configure the IS-IS default route

Step	Command	Description
Enter the global configuration mode	**configure terminal**	-
Enter the IS-IS configuration mode	**router isis** [*area-tag*]	-
Enter the IS-IS IPv6 address family configuration mode	**address-family ipv6 unicast**	-
Configure the IS-IS to release he default route	**default-information originate**	Mandatory By default, the default route is not released.

Configure IPV6 IS-IS Routing Redistribution

The routing redistribution can be used to introduce the routing information of other routing protocols to the IS-IS protocol. This enables the interconnection between the autonomous system of the IS-IS protocol and the autonomous system of other routing protocols or the routing area. When the external routing is introduced, the routing introduction policy and the routing layer attribute after introduction are specified.

Table 9-7Configure the IPV6 IS-IS routing redistribution

Step	Command	Description
Enter the global configuration mode	**configure terminal**	-
Enter the IS-IS configuration mode	**router isis** [*area-tag*]	-
Enter the IS-IS IPv6 address family configuration mode	**address-family ipv6 unicast**	-

Step	Command	Description
Configure the IS-IS routing redistribution	**redistribute** *protocol* [*protocol-id*] [**level-1** / **level-1-2** / **level-2** / **metric** *metric-value* / **metric-type** { **external** | **internal** } / **route-map** *route-map-name* / **match** *route-sub-type*]	Mandatory By default, information of other routing protocols are not redistributed.

9.2.4. Configure IPv6 IS-IS Routing Control

Configuration Condition

Before configuring the IPV6 IS-IS routing feature, first complete the following tasks:

- Configure the IP address of the interface to enable the neighboring nodes to be reachable at the network layer.
- Enable the IPV6 IS-IS protocol.

Configure IPV6 IS-IS Metric Style

Initially, the IS-IS only has the narrow metric style. When the narrow metric style is used, the maximum metric value is 63. With the expansion of the network scale, the metric style cannot satisfy the requirements. Therefore, the wide metric style emerges whose metric value can reach 16777214. Devices use different metric styles cannot advertise and learn the routing information from each other. To realize the transition between the two metric styles, the configuration method for the transition metric style is provided.

The wide metric style is recommended.

Table 9-8Configure the IPV6 IS-IS metric style

Step	Command	Description
Enter the global configuration mode	**configure terminal**	-
Enter the IS-IS configuration mode	**router isis** [*area-tag*]	-
Configure the interface metric style	**metric-style** {**narrow** | **narrow transition** | **transition** | **wide** | **wide transition**} [**level-1** | **level-1-2** | **level-2**]	Optional By default, the narrow metric style is used.

Configure IPV6 IS-IS Interface Metric Value

When the IS-IS protocol is enabled on the interface, the IS-IS routing metric is the global metric value. The following command can be used to specify a metric value for each interface.

Table 9-9 Configure the interface metric value

Step	Command	Description
Enter the global configuration mode	**configure terminal**	-
Enter the IS-IS configuration mode	**router isis** [*area-tag*]	-
Configure the IS-IS global metric value	**metric** *metric-value* **[level-1 | level-2]**	Optional By default, the global metric value is 10.
Return to the global configuration mode	**exit**	-
Enter the interface configuration mode	**interface** *interface-name*	-
Configure the interface metric value	**isis ipv6 metric** {*metric-value* **| maximum**} [**level-1 | level-2**]	Optional By default, the global metric value is used.

Configure IPV6 IS-IS Administrative Distance

The system chooses the primary routing based on the administrative distance. The smaller the administrative distance is, the higher priority the routing has.

Table 9-10 Configure the IPV6 IS-IS administrative distance

Step	Command	Description
Enter the global configuration mode	**configure terminal**	-
Enter the IPV6 IS-IS configuration mode	**router isis** [*area-tag*]	-

Step	Command	Description
Enter the IS-IS IPv6 address family configuration mode	**address-family ipv6 unicast**	-
Configure the IS-IS routing administrative distance	**distance** *distance-value*	Optional By default, the administrative distance is 115.

Configure IPV6 IS-IS Route Summary

The route summary summarizes multiple pieces of routing information as a piece of routing information. After the route summary is configured for the IS-IS, the number of advertisements to the subnet reduces effectively and the link status database and routing table size reduce. This effectively saves the memory and CPU resources. This configuration is generally applied to the Level-1-2 edge device, reducing the routing information of layer advertisement.

Table 9-11 Configure the IPV6 IS-IS route summary

Step	Command	Description
Enter the global configuration mode	**configure terminal**	-
Enter the IS-IS configuration mode	**router isis** [*area-tag*]	-
Enter the IPv6 address family configuration mode	**address-family ipv6 unicast**	-
Configure the IS-IS route summary	**summary-prefix** *prefix-value* [**metric** *metric-value* **/ route-type** {**internal | external**} **/ metric-type {internal | external**} **/ tag** *tag-value* **/ not-advertise / level-1 / level-2 / level-1-2]**	Mandatory By default, the route summary is not performed.

Configure the maximum number of load-balanced routes for the IPV6 IS-IS

There are multiple paths of the same cost to the same destination IP address. These ECMP (equal cost multipath routing) can improve the link utilization rate. The user can control the maximum number of the IS-IS ECMPs.

Table 9-12 Configure the maximum number of load-balanced routes for the IPV6 IS-IS

Step	Command	Description
Enter the global configuration mode	**configure terminal**	-
Enter the IS-IS configuration mode	**router isis** [*area-tag*]	-
Enter the IPv6 address family configuration mode	**address-family ipv6 unicast**	-
Configure the maximum number of load-balanced routes for the IS-IS	**maximum-paths** *max-number*	Optional By default, the maximum number of paths for load balancing is 4.

Configure IPV6 IS-IS Inter-layer Route Leakage

By default, the IS-IS only leak the Level-1 routing to the Level-2, but the Level-1 area cannot know the routing of the Level-2 area. The inter-layer route leakage can be configured to introduce the Level-2 routing to the Level-1 area. When configuring the inter-layer route leakage, the routing policy can be specified to only leak the route that matches the condition.

Table 9-13 Configure the IPV6 IS-IS inter-layer route leakage

Step	Command	Description
Enter the global configuration mode	**configure terminal**	-
Enter the IS-IS configuration mode	**router isis** [*area-tag*]	-
Enter the IS-IS IPv6 address family configuration mode	**address-family ipv6 unicast**	-
Configure the route leakage between the IS-IS layer	**propagate** { **level-1 into level-2** | **level-2 into level-1** } [**distribute-list** *access-list-name* | **route-map** *route-map-name*]	Mandatory By default, the Level-1 leaks its route to the Level-2.

Configure IPV6 IS-IS ATT-bit

In the Level-1-2 device, the ATT-bit is used to inform other nodes whether this node has the connection to other areas. If yes, the ATT-bit will be set to 1 automatically and other nodes will generate a default route to this node. This increases the service load of this node. To avoid this situation, the ATT-bit can be forcibly set to 0.

Table 9-14 Configure the IPV6 IS-IS ATT-bit

Step	Command	Description
Enter the global configuration mode	**configure terminal**	-
Enter the IS-IS configuration mode	**router isis** [*area-tag*]	-
Enter the IS-IS IPv6 address family configuration mode	**address-family ipv6 unicast**	-
Configure the IS-IS ATT-bit	**set-attached-bit** { **on** | **off** }	Mandatory By default, the ATT-bit is set based on whether the node is connected to other areas.

9.2.5. Configure IPV6 IS-IS Network Optimization

Configuration Condition

Before configuring the IPV6 IS-IS adjustment and optimization, first complete the following tasks:

- Configure the IP address of the interface to enable the neighboring nodes to be reachable at the network layer.
- Enable the IPV6 IS-IS protocol.

Configure IPV6 IS-IS Interface Priority

The IS-IS chooses a node on the broadcast link as the DIS node. The DIS node sends the CSNP packet periodically to synchronize the link status database on the entire network. The Level-1 and Level-2 select the DIS node respectively. The interface with the highest priority is selected as the DIS node. The node with the large MAC address is selected as the DIS node for the nodes with the same priority.

Table 9-15 Configure the IPV6 IS-IS interface priority

Step	Command	Description
Enter the global configuration mode	**configure terminal**	-
Enter the interface configuration mode	**interface** *interface-name*	-
Configure the IS-IS interface priority	**isis priority** *priority-value* [**level-1** | **level-2**]	Optional By default, the interface priority is 64.

Configure IPV6 IS-IS Passive Interface

The passive interface does not receive and transmit the IS-IS protocol packet, but still releases the directly connected network routing information of this interface. The IS-IS can reduce the bandwidth and CPU handling time through configuring the passive interface. Based on this configuration, the IS-IS can be specified to only release the directly connected network routing information of the passive interfaces and not release the directly connected network routing information of the non-passive interfaces.

Table 9-16 Configure the IPV6 IS-IS passive interface

Step	Command	Description
Enter the global configuration mode	**configure terminal**	-
Enter the IS-IS configuration mode	**router isis** [*area-tag*]	-
Configure the IS-IS passive interface	**passive-interface** *interface-name*	Mandatory By default, the IS-IS does not have the passive interface.
Configure the IS-IS only to release the routing information of the passive interface	**advertise-passive-only**	Optional By default, the directly connected network routing information of the interface enabled with the IS-IS protocol is released.

Configure IPV6 IS-IS Hello Packet Parameter

1. Configure the Hello packet delivery interval.

The interface enabled with the IS-IS protocol will send the Hello packet to keep the neighboring relationship with the neighboring devices. The smaller delivery interval of the Hello packet is, the faster the network convergence is. However, more bandwidth will be occupied.

Table 9-17 Configure delivery interval for the Hello packet

Step	Command	Description
Enter the global configuration mode	**configure terminal**	-
Enter the interface configuration mode	**interface** *interface-name*	-
Configure the Hello packet delivery interval on the interface	**isis hello-interval** { *interval* | **minimal** } [**level-1** | **level-2**]	Optional By default, the delivery interval of the Hello packet is 10s.

1. Configure the number of invalid Hello packets.

The IPV6 IS-IS calculates the neighbor relationship retention time based on the number of invalid Hello packets and informs the retention time to the neighboring device. If the neighboring device does not receive the Hello packet from this device during this period, the neighbor relationship is invalid and the routing calculation will be recalculated.

Table 9-18 Configure the number of invalid Hello packets

Step	Command	Description
Enter the global configuration mode	**configure terminal**	-
Enter the interface configuration mode	**interface** *interface-name*	-
Configure the number of invalid Hello packets on the interface	**isis hello-multiplier** *multiplier* [**level-1** | **level-2**]	Optional By default, the number of invalid Hello packets is 3.

1. Configure to cancel the Hello packet padding function.

If MTU values on the interface at both sides of the link are inconsistent, as a result, smaller packets can be transmitted but larger packets cannot be transmitted. To avoid such situation, the IS-IS adopts the padding Hello packet to the interface MTU value to make the neighbor relationship cannot be established. However, this method wastes the bandwidth. In actual network, there is no need to configure the padding Hello packet. Only the small Hello packets are transmitted.

Table 9-19 Configure to cancel the Hello packet padding function

Step	Command	Description
Enter the global configuration mode	**configure terminal**	-
Enter the interface configuration mode	**interface** *interface-name*	-
Cancel the Hello packet padding function	**no isis hello padding**	Mandatory By default, the Hello packet padding function is enabled.

Configure IPV6 IS-IS LSP Packet Parameter

1. Configure the maximum survival time for the LSP packet.

Each LSP packet has a maximum survival time. When the survival time of the LSP packet reduces to 0, the LSP packet will be deleted from the link status database. The maximum survival time of the LSP packet must be larger than the LSP packet refresh interval.

Table 9-20 Configure the IPV6 IS-IS LSP packet parameter

Step	Command	Description
Enter the global configuration mode	**configure terminal**	-
Enter the IS-IS configuration mode	**router isis** [*area-tag*]	-
Configure the maximum survival time for the LSP packet	**max-lsp-lifetime** *life-time*	Optional By default, the maximum survival time of the LSP is 1200s.

1. Configure the LSP packet refresh interval.

The IS-IS protocol advertises and learns the routing through interacting each LSP packets. The nodes save the received LSP packets in the link status database. Each LSP packet has a maximum survival time and each node needs to refresh its LSP packet periodically to prevent the LSP packet maximum survival time reducing to 0 and keep the LSP packet in the entire area synchronization. Reducing the LSP packet delivery interval can accelerate the network convergence speed, but will occupy more bandwidth.

Table 9-21 Configure the LSP packet update packet

Step	**Command**	**Description**
Enter the global configuration mode	**configure terminal**	-
Enter the IS-IS configuration mode	**router isis** [*area-tag*]	-
Configure the LSP packet refresh interval	**lsp-refresh-interval** *refresh-interval*	Optional By default, the packet refresh interval for the periodical packet delivery is 900s.

1. Configure the LSP packet generation interval.

Periodical refresh will generate new LSP packet. Besides, the interface status changes and network status changes will also trigger new LSP packet generation. To prevent frequently generated LSP packets occupying too much CPU resources, the user can configure the minimum LSP packet generation interval.

Table 9-22 Configure the LSP packet generation interval

Step	**Command**	**Description**
Enter the global configuration mode	**configure terminal**	-
Enter the IS-IS configuration mode	**router isis** [*area-tag*]	-
Configure the LSP packet generation interval	**lsp-gen-interval** [**level-1** | **level-2**] *max-interval* [*initial-interval* [*secondary-interval*]]	Optional By default, the LSP packet generation interval is 50 ms.

1. Configure the LSP packet delivery interval.

Every generated LSP packet will be delivered on the interface. To avoid frequently generated LSP packet will greatly occupy the interface bandwidth. Each interface is configured with the minimum delivery interval of the LSP packet.

Table 9-23 Configure the LSP packet delivery interval

Step	Command	Description
Enter the global configuration mode	**configure terminal**	-
Enter the interface configuration mode	**interface** *interface-name*	-
Configure the LSP packet delivery interval	**isis lsp-interval** *min-interval*	Optional By default, the delivery interval of the LSP packet is 33 ms.

1. Configure the LSP packet retransmission time.

On the point-to-point link, the IS-IS sends the LSP packet and the then requires the peer end to send the PSNP acknowledgement message. If the IS-IS does not receive the acknowledgement message, the IS-IS will send the LSP packet again. The time waiting the acknowledgement message is the LSP packet retransmission interval. The retransmission interval can be set as required by the user to avoid LSP packet retransmission when the acknowledgement message is not received due to large delay.

Table 9-24 Configure the LSP packet retransmission time

Step	Command	Description
Enter the global configuration mode	**configure terminal**	-
Enter the interface configuration mode	**interface** *interface-name*	-
Configure the LSP packet retransmission time	**isis retransmit-interval** *interval* [**level-1** | **level-2**]	Optional By default, the retransmission time is 5s.

1. Configure the LSP MTU value.

The IS-IS protocol packet cannot perform automatic fragmentation. In order not to affecting normal LSP packet spread, the maximum length of the LSP packet in a routing domain cannot exceed the minimum MTU value on the IS-IS interfaces of all devices. Therefore, when the interface MTU values are inconsistent on devices in the routing domain, it is recommended that the maximum length of the LSP packet is set uniformly.

Table 9-25 Configure the LSP MTU value

Step	Command	Description
Enter the global configuration mode	**configure terminal**	-
Enter the IS-IS configuration mode	**router isis** [*area-tag*]	-
Configure the LSP packet MTU value	**lsp-mtu** *mtu-size* [**level-1** | **level-2**]	Optional By default, the MTU value of the LSP packet is 1492 bytes.

Configure IPV6 IS-IS SNP Packet Parameter

1. Configure the CSNP packet delivery interval.

The selected nodes on the broadcast link need to send the CSNP packet periodically to synchronize the link status database on the entire network. The CSNP packet delivery interval is adjusted based on the actual situation.

Table 9-26 Configure the CSNP packet delivery interval

Step	Command	Description
Enter the global configuration mode	**configure terminal**	-
Enter the interface configuration mode	**interface** *interface-name*	-
Configure the CSNP packet delivery interval	**isis csnp-interval** *interval* [**level-1** | **level-2**]	Optional By default, the CSNP packet delivery interval is 10s.

1. Configure the PSNP packet delivery interval

On the broadcast link, the PSNP packet synchronizes the link status database on the entire network. On the point-to-point link, the PSNP packet confirms the received LSP packet. To avoid a large number of PSNP packets being delivered over the interface. A minimum delivery interval is set for the PSNP packet and the user can change the interval dynamically. The PSNP packet delivery interval cannot be set to a too large value. If the packet delivery interval is set to a too large value, the link status database synchronization on the entire network will be affected for the broadcast link, and the LSP packet may be redelivered caused by not timely receiving the acknowledgment message for the point-to-point link.

Table 9-27 Configure the PSNP packet delivery interval

Step	Command	Description
Enter the global configuration mode	**configure terminal**	-
Enter the interface configuration mode	**interface** *interface-name*	-
Configure the PSNP packet delivery interval	**isis psnp-interval** *min-interval* [**level-1** | **level-2**]	Optional By default, the PSNP packet delivery interval is 2s.

Configure IPV6 IS-IS SPF Calculation Interval

The IS-IS link status database changes will trigger the SPF routing calculation. Frequent SPF calculation will consume a mass of CPU resources and user can configure the SPF calculation interval.

Table 9-28 Configure the IPV6 IS-IS SPF calculation interval

Step	Command	Description
Enter the global configuration mode	**configure terminal**	-
Enter the IS-IS configuration mode	**router isis** [*area-tag*]	-
Enter the IS-IS IPv6 address family configuration mode	**address-family ipv6 unicast**	-
Configure the IS-IS SPF calculation interval	**spf-interval** [**level-1** | **level-2**] *maximum-interval* [*min-initial-delay* [*min-second-delay*]]	Optional By default, *maximum-interva* l is 10s, *min-initial-delay* is 50ms, *min-second-delay* is 200ms.

Configure Maximum Number of Areas for IPV6 IS-IS

Multiple area IP addresses can be configured in an IS-IS process. Multiple area addresses are mainly applied in the following two situations that multiple Level-1 areas are combined as a Level-1 area, or a Level-1 area is divided into multiple Level-1 areas.

Table 9-29 Configure the maximum number of areas for the IPV6 IS-IS

Step	Command	Description
Enter the global configuration mode	**configure terminal**	-
Enter the IS-IS configuration mode	**router isis** [*area-tag*]	-
Configure the maximum number of areas for the IS-IS	**max-area-addresses** *max-number*	Optional By default, the maximum number of the area addresses is 3.

Note:

- This configuration must be consistent in the entire IS-IS Level-1 routing domain. Otherwise, the Level-1 neighbor cannot be established normally. The Level-2 neighbor is not affected.

Configure IPV6 IS-IS Host Name Mapping

The IS-IS uniquely identifies a intermediate system using the system ID with a fixed length of 6 bytes. When viewing the system information such as the neighbor relationship and link status database, the system ID cannot enable the user to visually associate the system ID with the host name. The IS-IS supports the mapping between the system ID and the host name to enable the user to view the system information more visually and conveniently. The IS-IS host name mapping can be configured in the following two methods:

1. Configure the IS-IS static host name mapping.

The IS-IS static host name mapping is manually established by the user between the system ID and the host name for the remote device.

Table 9-30 Configure the IS-IS static host name mapping

Step	Command	Description
Enter the global configuration mode	**configure terminal**	-
Enter the IS-IS configuration mode	**router isis** [*area-tag*]	-
Configure the IS-IS static hostname mapping	**hostname static** *system-id host-name*	Mandatory

1. Configure the IS-IS dynamic host name mapping.

The static host name mapping requires the user to configure the system ID and host name mapping of other devices on each device in the network, which has a heavy workload. The dynamic host name mapping only configures the host name for each device, and other devices in the network can learn the host name of the device when the host name advertisement function is enabled.

Table 9-31 Configure the IS-IS dynamic host name mapping

Step	Command	Description
Enter the global configuration mode	**configure terminal**	-
Enter the IS-IS configuration mode	**router isis** [*area-tag*]	-
Configure the IS-IS dynamic host name mapping	**hostname dynamic** { *host-name* | **area-tag** | **recv-only** | **system-name** }	Mandatory By default, only the host names advertised by other devices are learnt.

Configure IPV6 IS-IS Interface to Be Added to Mesh Group

When the IS-IS interface is not added to the mesh group, the LSP packet received from an interface will be sent out on all the other IS-IS interfaces. This results in great bandwidth waste in a full mesh connected network. To avoid this situation, several IS-IS interfaces can be added to a mesh group. When an interface receives the LSP packet, it only sends the LSP packet out to the interface that is not in the same mesh group with this interface.

Table 9-32 Configure the IS-IS interface to be added to the mesh group

Step	Command	Description
Enter the global configuration mode	**configure terminal**	-
Enter the interface configuration mode	**interface** *interface-name*	-
Configure the IS-IS interface to be added to the mesh group	**isis mesh-group** { *group-number* | **blocked** }	Mandatory By default, the IS-IS interface is not added to the mesh group.

Note:

- The **isis mesh-group blocked** command can be used to configure the interface as the obstructive interface. The obstructive interface will not send the LSP packet actively and only send the LSP packet when receives the LSP request.

9.2.6. Configure IPV6 IS-IS Network Authentication

Configuration Condition

Before configuring the IPV6 IS-IS network authentication, first complete the following tasks:

- Configure the IPv6 address of the interface to enable the neighboring nodes to be reachable at the network layer.
- Enable the IPV6 IS-IS protocol.

Configure IPV6 IS-IS Neighboring Authentication

When the neighbor relationship authentication is enabled for the IS-IS, the authentication information will be added to the delivered Hello packet and the received Hello packet will be authenticated. If the authentication fails, the neighbor relationship will not be established. This can prevent the neighbor relationship being established with the unreliable devices.

Table 9-33 Configure the IS-IS neighboring authentication

Step	Command	Description
Enter the global configuration mode	**configure terminal**	-
Enter the interface configuration mode	**interface** *interface-name*	-
Configure the Hello packet authentication mode	**isis authentication mode** { **md5** | **text** } [**level-1** | **level-2**]	Mandatory By default, the authentication function is not enabled.
Configure the Hello packet authentication password	**isis authentication key** { **0** | **7** } *password* [**level-1** | **level-2**]	Either By default, the authentication password is not configured. The authentication password can be configured using the password chain. For details about the password chain configuration, refer to the password chain configuration chapter in the configuration manual.
	isis authentication key-chain *key-chain-name* [**level-1** | **level-2**]	

Configure IPV6 IS-IS Route Authentication

When the routing information authentication is enabled for the IS-IS, the authentication information will be added to the LSP and SNP packets and the received LSP and SNP packets will be authenticated. If the authentication fails, the packet will be dropped directly. This can prevent the unreliable routing information spreading to the IS-IS network.

Table 9-34 Configure the IS-IS route authentication

Step	Command	Description
Enter the global configuration mode	**configure terminal**	-
Enter the IS-IS configuration mode	**router isis** [*area-tag*]	-
Configure the authentication mode of the routing information packet	**authentication mode** { **md5** | **text** } [**level-1** | **level-2**]	Mandatory By default, the authentication function is not enabled.
Configure the authentication password of the routing information packet	**authentication key** { **0** | **7** } *password* [**level-1** | **level-2**]	Either By default, the authentication password is not configured. The authentication password can be configured using the password chain. For details about the password chain configuration, refer to the password chain configuration chapter in the configuration manual.
	authentication key-chain *key-chain-name* [**level-1** | **level-2**]	

9.2.7. Configure IPv6 IS-IS Fast Re-routing

Configuration Conditions

Before configuring IS-IS fast re-routing, first complete the following task:

- Configure the IPv6 protocol of the interface, making the neighboring node reachable at the network layer.
- Enable the IPv6 IS-IS protocol.

Configure IPv6 IS-IS Fast Re-routing

In IPv6 IS-IS network, due to link or device failure, the packet passing through the failure point will be discarded or generate a loop. The traffic interruption caused by this will continue until the protocol reconverges, which often lasts for several seconds. In order to reduce the traffic interruption time, the IS-IS fast rerouting can be configured. By applying the route map, the backup next hop can be set for the successfully matched route. Once the main link fails, the traffic passing through the failed link will be immediately switched to the backup link, so as to realize fast switching.

Table 9-35 Configure IPv6 IS-IS fast re-routing

Step	Command	Description
Enter the global configuration mode	**configure terminal**	-
Enter the IS-IS configuration mode	**router isis** [*area-tag*]	-
Enter IS-IS IPv6 address family configuration mode	**address-family ipv6 unicast**	-
Configure IPv6 ISIS to enable the fast re-routing function	**fast-reroute route-map** *route-map-name*	Mandatory By default, do not enable the IPv6 IS-IS fast re-routing function.
Configure IPv6 IS-IS to enable the dynamic fast re-routing function	**fast-reroute loop-free-alternate** [**route-map** *route-map-name*]	Mandatory By default, do not enable the IPv6 IS-IS dynamic fast re-routing function.
Configure IPv6 IS-IS to enable the pic function	**pic**	Mandatory After enabling the pic function, enable the auto fast re-routing function. By default, do not enable the IPv6 IS-IS pic function.

Note:

- The IPv6 IS-IS fast reroute function is divided into static fast reroute and dynamic fast reroute.
- The static fast rerouting function needs to associate with the route-map, and set the next hop interface and address of the backup route in the route-map.

- At present, dynamic fast reroute only supports point-to-point network, that is, the network type of all outgoing interfaces of the device needs to be point-to-point. After configuring dynamic fast reroute, the device automatically calculates and sets the backup next hop interface and address. Dynamic fast rerouting can also be associated with route-map. Only the routes matching route map are set to back up the next hop interface and address.
- The various modes of enabling rerouting are mutually exclusive;

9.2.8. IPV6 IS-IS Monitoring and Maintaining

Table 9-36 The IPV6 IS-IS monitoring and maintaining

Command	Description
clear isis [**instance -null** | *area-tag*] **statistics** [*interface_name*]	Clear the statistics information of the IPV6 IS-IS protocol operation
clear isis [**instance -null** | *area-tag*] **process**	Restart the IPV6 IS-IS protocol process
show isis [**instance -null** | *area-tag*]	Display the IPV6 IS-IS process information
show isis instance { **-null** | *area-tag* } **bfd-sessions**	Display the BFD session information of the IPV6 IS-IS process
show isis [**instance** -null | *area-tag*] **database** [*lsp_id*] [**detail**] [**l1** / **l2**] [**level-1** / **level-2**] [**self**] [**verbose**]	Display the IPV6 IS-IS link status database information
show isis interface [*interface-name*] [**detail**]	Display the information of the IPV6 IS-IS protocol interface operation
show isis [**instance –null** | *area-tag*] **ipv6 reach-info**	Display the IPV6 IS-IS IPv4 subnet reachable information
show isis [**instance –null** | *area-tag*] **ipv6 route**	Display the IPV6 IS-IS IPv4 routing information
show isis [**instance –null** | *area-tag*] **ipv6 topology**	Display the IPV6 IS-IS IPv4 topology information
show isis [**instance – null** | *area-tag*] **is-reach-info** [**level-1** | **level-2**]	Display the IPV6 IS-IS neighboring node information

Command	Description
show isis [**instance –null** | *area-tag*] **mesh-groups**	Display the IPV6 IS-IS mesh group
show isis [**instance –null** | *area-tag*] **neighbors** [*interface-name*] [**detail**]	Display the IPV6 IS-IS neighbor information
show isis [**instance –null** | *area-tag*] **statistics** [*interface-name*]	Display the statistics information of the IPV6 IS-IS protocol operation
show isis router	Display the IPV6 IS-IS host name information

9.3. IPv6 IS-IS Typical Configuration Example

9.3.1. Configure IPv6 IS-IS Basic Function

Network Requirements

- Configure the IPv6 IS-IS protocol to realize the network interconnection between devices.
- Device1 is the Level-1 router and Device2 is the Level-1-2 router. Device1 and Device2 are in the same area, Area 10. Device3 is the Level-2 router in Area 20. Device2 connects the two areas.

Network Topology

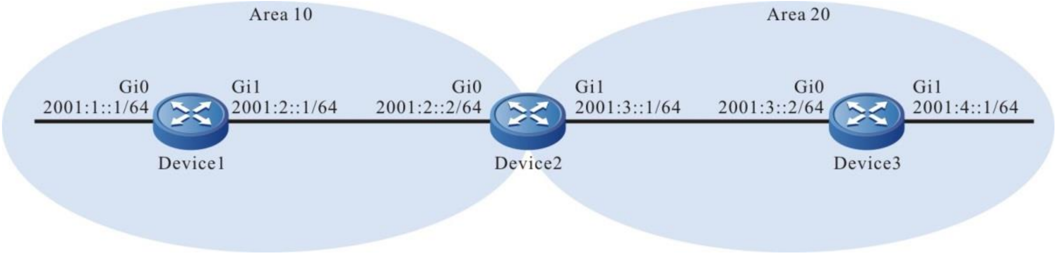

Figure 9–1 Networking of configuring the IPv6 IS-IS basic functions

Configuration Steps

Step 1: Configure the IPv6 address of the interfaces. (Omitted)

Step 2: Configure the IPv6 IS-IS and enable the process on the interface.

#Configure the IPv6 IS-IS process as 100, area number as 10, and type as Level-1 and enable the process on the interface of Device1.

```
Device1#configure terminal
Device1(config)#router isis 100
Device1(config-isis)#net 10.0000.0000.0001.00
Device1(config-isis)#is-type level-1
```

```
Device1(config-isis)#metric-style wide
Device1(config-isis)#address-family ipv6 unicast
Device1(config-isis-af)#multi-topology
Device1(config-isis-af)#exit-address-family
Device1(config-isis)#exit
Device1(config)#interface gigabitethernet 0
Device1(config-if-gigabitethernet0)#ipv6 router isis 100
Device1(config-if-gigabitethernet0)#exit
Device1(config)#interface gigabitethernet 1
Device1(config-if-gigabitethernet1)#ipv6 router isis 100
Device1(config-if-gigabitethernet1)#exit
```

#Configure the IPv6 IS-IS process as 100, area number as 10, and type as Level-1-2 and enable the process on the interface of Device2.

```
Device2#configure terminal
Device2(config)#router isis 100
Device2(config-isis)#net 10.0000.0000.0002.00
Device2(config-isis)#metric-style wide
Device2(config-isis)#address-family ipv6 unicast
Device2(config-isis-af)#multi-topology
Device2(config-isis-af)#exit-address-family
Device2(config-isis)#exit
Device2(config)#interface gigabitethernet 0
Device2(config-if-gigabitethernet0)#ipv6 router isis 100
Device2(config-if-gigabitethernet0)#exit
Device2(config)#interface gigabitethernet 1
Device2(config-if-gigabitethernet1)#ipv6 router isis 100
Device2(config-if-gigabitethernet1)#exit
```

#Configure the IPv6 IS-IS process as 100, area number as 20, and type as Level-2 and enable the process on the interface of Device3.

```
Device3#configure terminal
Device3(config)#router isis 100
Device3(config-isis)#net 20.0000.0000.0003.00
Device3(config-isis)#is-type level-2
Device3(config-isis)#metric-style wide
Device3(config-isis)#address-family ipv6 unicast
Device3(config-isis-af)#multi-topology
Device3(config-isis-af)#exit-address-family
Device3(config-isis)#exit
```

```
Device3(config)#interface gigabitethernet0
Device3(config-if-gigabitethernet0)#ipv6 router isis 100
Device3(config-if-gigabitethernet0)#exit
Device3(config)#interface gigabitethernet1
Device3(config-if-gigabitethernet1)#ipv6 router isis 100
Device3(config-if-gigabitethernet1)#exit
```

Step 3: Check the result.

#View the IPv6 IS-IS neighboring information of Device1.

```
Device1#show isis neighbors
IS-IS Instance 100 Neighbors (Counter 1):
Type   System ID      Interface                  State Holdtime Level IETF-NSF Priority
Circuit ID
L1-LAN 0000.0000.0002 gigabitethernet1                Up    25 sec   L1    capable  64
0000.0000.0001.02
```

#View the IPv6 IS-IS neighboring information on Device2.

```
Device2#show isis neighbors
IS-IS Instance 100 Neighbors (Counter 2):
Type   System ID      Interface                  State Holdtime Level IETF-NSF Priority Circuit
ID
L1-LAN 0000.0000.0001 gigabitethernet0                Up    8 sec    L1    capable  64
0000.0000.0001.02
L2-LAN 0000.0000.0003 gigabitethernet1                Up    7 sec    L2    capable  64
0000.0000.0003.01
```

Device2 builds the IPv6 IS-IS neighbor with Device1 and Device3, respectively.

#View the IPv6 IS-IS neighboring information of Device3.

```
Device3#show isis neighbors
IS-IS Instance 100 Neighbors (Counter 1):
Type   System ID      Interface                  State Holdtime Level IETF-NSF Priority
Circuit ID
L2-LAN 0000.0000.0002 gigabitethernet0                Up    23 sec   L2    capable  64
0000.0000.0003.01
```

#View the routing information of Device1.

```
Device1#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
     U - Per-user Static route
     O - OSPF, OE-OSPF External, M - Management
```

```
i   ::/0 [115/10]
    via fe80::201:7aff:fe5e:6d45, 00:03:17, gigabitethernet1
L   ::1/128 [0/0]
    via ::, 16:03:18, lo0
C   2001:1::/64 [0/0]
    via ::, 00:20:11, gigabitethernet0
L   2001:1::1/128 [0/0]
    via ::, 00:20:10, lo0
C   2001:2::/64 [0/0]
    via ::, 00:19:50, gigabitethernet1
L   2001:2::1/128 [0/0]
    via ::, 00:19:49, lo0
i   2001:3::/64 [115/20]
    via fe80::201:7aff:fe5e:6d45, 00:06:48, gigabitethernet1

Device1#show isis ipv6 route
IS-IS Instance 100, VRF Kernel, IPv6 routes table (Counter 4):
L1 ::/0, flags none, metric 10, from learned, installed
    via fe80::201:7aff:fe5e:6d45, gigabitethernet1, neighbor 0000.0000.0002
L1 2001:1::/64, flags none, metric 10, from network connected
    via ::, gigabitethernet0
L1 2001:2::/64, flags none, metric 10, from network connected
    via ::, gigabitethernet1
L1 2001:3::/64, flags none, metric 20, from learned, installed
    via fe80::201:7aff:fe5e:6d45, gigabitethernet1, neighbor 0000.0000.0002
```

A default routing is in the routing table of Device1 and the next hop is Device2.

#View the routing information of Device2.

```
Device2#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
      U - Per-user Static route
      O - OSPF, OE-OSPF External, M - Management

L   ::1/128 [0/0]
    via ::, 1d:12:42:31, lo0
i   2001:1::/64 [115/20]
    via fe80::201:7aff:fe61:7a24, 00:08:32, gigabitethernet0
```

```
C   2001:2::/64 [0/0]
    via ::, 23:41:09, gigabitethernet0
L   2001:2::2/128 [0/0]
    via ::, 23:41:06, lo0
C   2001:3::/64 [0/0]
    via ::, 17:39:09, gigabitethernet1
L   2001:3::1/128 [0/0]
    via ::, 17:39:06, lo0
i   2001:4::/64 [115/20]
    via fe80::2212:1ff:fe01:101, 00:04:52, gigabitethernet1

Device2#show isis ipv6 route
IS-IS Instance 100, VRF Kernel, IPv6 routes table (Counter 4):
L1 2001:1::/64, flags none, metric 20, from learned, installed
    via fe80::201:7aff:fe61:7a24, gigabitethernet0, neighbor 0000.0000.0001
L1 2001:2::/64, flags none, metric 10, from network connected
    via ::, gigabitethernet0
L1 2001:3::/64, flags none, metric 10, from network connected
    via ::, gigabitethernet1
L2 2001:4::/64, flags none, metric 20, from learned, installed
    via fe80::2212:1ff:fe01:101, gigabitethernet1, neighbor 0000.0000.0003
```

Device2 contains the Level-1 and Level-2 routing.

#View the routing information of Device3.

```
Device3#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
      U - Per-user Static route
      O - OSPF, OE-OSPF External, M - Management

L   ::1/128 [0/0]
    via ::, 00:00:12, lo0
i   2001:1::/64 [115/30]
    via fe80::201:7aff:fe5e:6d46, 00:00:12, gigabitethernet0
i   2001:2::/64 [115/20]
    via fe80::201:7aff:fe5e:6d46, 00:00:12, gigabitethernet0
C   2001:3::/64 [0/0]
    via ::, 00:00:12, gigabitethernet0
```

```
L   2001:3::2/128 [0/0]
    via ::, 00:00:12, lo0
C   2001:4::/64 [0/0]
    via ::, 00:00:12, gigabitethernet1
L   2001:4::1/128 [0/0]
    via ::, 00:00:12, lo0

Device3#show isis ipv6 route
IS-IS Instance 100, VRF Kernel, IPv6 routes table (Counter 4):
L2 2001:1::/64, flags none, metric 30, from learned, installed
    via fe80::201:7aff:fe5e:6d46, gigabitethernet0, neighbor 0000.0000.0002
L2 2001:2::/64, flags none, metric 20, from learned, installed
    via fe80::201:7aff:fe5e:6d46, gigabitethernet0, neighbor 0000.0000.0002
L2 2001:3::/64, flags none, metric 10, from network connected
    via ::, gigabitethernet0
L2 2001:4::/64, flags none, metric 10, from network connected
    via ::, gigabitethernet1
```

Device3 learns the Level-1 route and the Level-1 leaks the route to Level-2 by default.

Note:

- The metric type is the narrow metric by default. The wide metric is recommended.
- The IPv6 IS-IS entity attribute is Level-1-2 by default.

9.3.2. Configure IPv6 IS-IS Static Fast Re-routing

Network Requirements

- All devices are configured with the IPv6 IS-IS protocol.
- Static fast rerouting is enabled between Device1 and Device3. When the line fails, the service can be quickly switched to Device2 for communication.

Network Topology

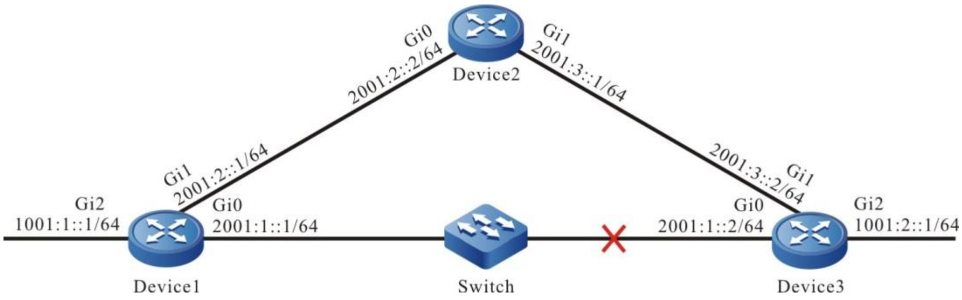

Figure 9-2 Networking of configuring IPv6 IS-IS static fast re-routing

Configuration Steps

Step 1: Configure the IPv6 address of the interfaces. (Omitted)

Step 2: Configure the IPv6 IS-IS and enable the process on the interface.

#Configure the IPv6 IS-IS process as 100, area number as 10, and type as Level-1 and enable the process on the interface of Device1.

```
Device1#configure terminal
Device1(config)#router isis 100
Device1(config-isis)#net 10.0000.0000.0001.00
Device1(config-isis)#is-type level-2
Device1(config-isis)#metric-style wide
Device1(config-isis)#exit
Device1(config)#interface gigabitethernet 0
Device1(config-if-gigabitethernet0)#ipv6 router isis 100
Device1(config-if-gigabitethernet0)#exit
Device1(config)#interface gigabitethernet 1
Device1(config-if-gigabitethernet1)#ipv6 router isis 100
Device1(config-if-gigabitethernet1)#exit
```

#Configure the IPv6 IS-IS process as 100, area number as 10, and type as Level-2 and enable the process on the interface of Device2.

```
Device2#configure terminal
Device2(config)#router isis 100
Device2(config-isis)#net 10.0000.0000.0002.00
Device2(config-isis)#is-type level-2
Device2(config-isis)#metric-style wide
Device2(config-isis)#exit
Device2(config)#interface gigabitethernet 0
Device2(config-if-gigabitethernet0)#ipv6 router isis 100
Device2(config-if-gigabitethernet0)#exit
Device2(config)#interface gigabitethernet 1
```

```
Device2(config-if-gigabitethernet1)#ipv6 router isis 100
Device2(config-if-gigabitethernet1)#exit
```

#Configure the IPv6 IS-IS process as 100, area number as 10, and type as Level-2 and enable the process on the interface of Device3.

```
Device3#configure terminal
Device3(config)#router isis 100
Device3(config-isis)#net 10.0000.0000.0003.00
Device3(config-isis)#is-type level-2
Device3(config-isis)#metric-style wide
Device3(config-isis)#exit
Device3(config)#interface gigabitethernet0
Device3(config-if-gigabitethernet0)#ipv6 router isis 100
Device3(config-if-gigabitethernet0)#exit
Device3(config)#interface gigabitethernet1
Device3(config-if-gigabitethernet1)#ipv6 router isis 100
Device3(config-if-gigabitethernet1)#exit
Device3(config-if-gigabitethernet2)#ipv6 router isis 100
Device3(config-if-gigabitethernet2)#exit
```

Step 3: On the gigabitethernet0 interface of Device3, configure the echo function of ipv6 bfd.

```
Device3(config)#interface gigabitethernet0
Device3(config-if-gigabitethernet0)# ipv6 bfd echo
Device3(config-if-gigabitethernet0)#exit
```

Step 4: Configure the routing policy.

#Configure Device1, configure route-map, and call ACL to match only 1001:2::1/64, while other network segments will be filtered out. The routing application matching the match rule backs up the next hop interface gigabitethernet1, and the next hop address 2001:2::2.

```
Device1(config)#ipv6 access-list extended 7001
Device1(config-v6-list)#permit ipv6 1001:2::1/64 any
Device1(config-v6-list)#exit
Device1(config)#route-map ipv6frr_isis
Device1(config-route-map)#match ipv6 address 7001
Device1(config-route-map)#set ipv6 fast-reroute backup-interface gigabitethernet
1 backup-nexthop 2001:2::2
Device1(config-route-map)#exit
```

Step 5: Configure the static fast re-routing.

```
Device1(config)#router isis 100
Device1(config-isis)#address-family ipv6 unicast
```

```
Device1(config-isis-af)#fast-reroute route-map ipv6frr_isis
Device1(config-isis-af)#exit-address-family
Device1(config-isis)#exit
```

Step 6: Check the result.

#View the IPv6 IS-IS route table of Device1.

```
Device1#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management
L   ::1/128 [0/0]
    via ::, 06:22:27, lo0
C   1001:1::/64 [0/0]
    via ::, 06:17:34, gigabitethernet2
L   1001:1::1/128 [0/0]
    via ::, 06:17:34, gigabitethernet2
i  1001:2::/64 [115/20]
    via fe80::201:7aff:fe92:e6b6, 00:10:32, gigabitethernet0
C   2001:1::/64 [0/0]
    via ::, 01:24:08, gigabitethernet0
L   2001:1::1/128 [0/0]
    via ::, 01:24:08, gigabitethernet0
C   2001:2::/64 [0/0]
    via ::, 06:16:52, gigabitethernet1
L   2001:2::1/128 [0/0]
    via ::, 06:16:52, gigabitethernet1
i  2001:3::/64 [115/20]
    via fe80::201:7aff:fe92:e6b6, 00:29:05, gigabitethernet0
    via fe80::ced8:1fff:fe10:7aae, 00:35:19, gigabitethernet1
```

It can be seen from the routing table that route 1001:2::/64 preferably communicates with the line between Device1 and Device3.

#View the IPv6 FRR routing table of Device1.

```
Device1#show ipv6 frr route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management
i  1001:2::/64 [115/4294967295]
    via 2001:2::2, 00:13:16, gigabitethernet1
```

You can see that the next hop of the frr route 1001:2::/64 is 2001:2::2, and the outgoing interface is gigabitethernet1.

#View the BFD session information of Device1.

```
Device1#show bfd session ipv6 detail
Total ipv6 session number: 1
OurAddr                      NeighAddr           LD/RD          State     Holddown
Interface
fe80::201:7aff:fe7d:1f1b         fe80::201:7aff:fe92:e6b6   1024/1024      UP       500
gigabitethernet0
Type:ipv6 direct  Mode:echo
Local Discriminator:369  Remote Discriminator:369
Local State:UP  Remote State:UP  Up for: 0h:15m:38s  Number of times UP:1
Send Interval:100ms  Detection time:500ms(100ms*5)
Local Diag:0  Demand mode:0  Poll bit:0
Registered modules:FIB_MGR
```

You can see that FIB_MGR is linked with BFD successfully, the session is established normally, and the session mode is echo.

#After the line between Device1 and Device3 fails, it can quickly detect and switch to Device2 for communication. Check the routing table of Device1. In the routing table, the outgoing interface to the destination network 1001:2::/64 has been switched to the backup interface gigabitethernet1.

```
Device1#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management
L   ::1/128 [0/0]
     via ::, 06:29:08, lo0
C   1001:1::/64 [0/0]
     via ::, 06:24:15, gigabitethernet2
L   1001:1::1/128 [0/0]
     via ::, 06:24:15, gigabitethernet2
i   1001:2::/64 [115/30]
     via fe80::ced8:1fff:fe10:7aae, 00:00:01, gigabitethernet1
i   2001:1::/64 [115/30]
     via fe80::ced8:1fff:fe10:7aae, 00:00:01, gigabitethernet1
C   2001:2::/64 [0/0]
     via ::, 06:23:33, gigabitethernet1
L   2001:2::1/128 [0/0]
     via ::, 06:23:33, gigabitethernet1
i   2001:3::/64 [115/20]
```

```
via fe80::ced8:1fff:fe10:7aae, 00:42:00, gigabitethernet1
```

9.3.3. Configure IPv6 IS-IS Dynamic Fast Re-routing

Network Requirements

- All devices are configured with the IPv6 IS-IS protocol.
- Dynamic fast rerouting is enabled between Device1 and Device3. When the line fails, the service can be quickly switched to Device2 for communication.

Network Topology

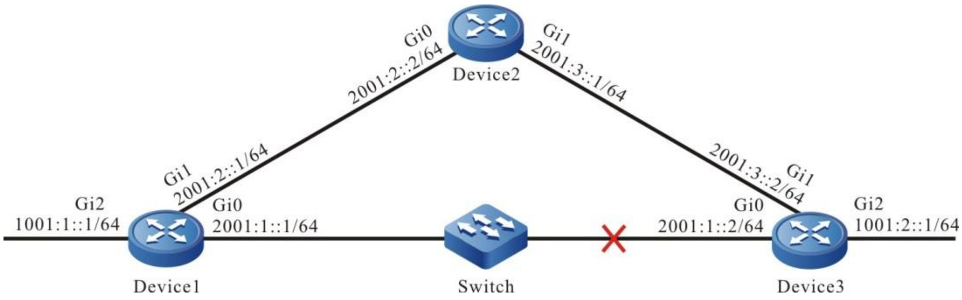

Figure 9-3 Networking of configuring IPv6 IS-IS dynamic fast re-routing

Configuration Steps

Step 1: Configure the IPv6 address of the interfaces. (Omitted)

Step 2: Configure the IPv6 IS-IS and enable the process on the interface.

#Configure the IPv6 IS-IS process as 100, area number as 10, and type as Level-2 and enable the process on the interface of Device1. Configure the interface network type as point-to-point.

```
Device1#configure terminal
Device1(config)#router isis 100
Device1(config-isis)#net 10.0000.0000.0001.00
Device1(config-isis)#is-type level-2
Device1(config-isis)#metric-style wide
Device1(config-isis)#exit
Device1(config)#interface gigabitethernet 0
Device1(config-if-gigabitethernet0)#ipv6 router isis 100
Device1(config-if-gigabitethernet0)# isis network point-to-point
Device1(config-if-gigabitethernet0)#exit
Device1(config)#interface gigabitethernet 1
Device1(config-if-gigabitethernet1)#ipv6 router isis 100
Device1(config-if-gigabitethernet1)# isis network point-to-point
Device1(config-if-gigabitethernet1)#exit
```

#Configure the IPv6 IS-IS process as 100, area number as 10, and type as Level-2 and enable the process on the interface of Device2. Configure the interface network type as point-to-point.

```
Device2#configure terminal
Device2(config)#router isis 100
```

```
Device2(config-isis)#net 10.0000.0000.0002.00
Device2(config-isis)#is-type level-2
Device2(config-isis)#metric-style wide
Device2(config-isis)#exit
Device2(config)#interface gigabitethernet 0
Device2(config-if-gigabitethernet0)#ipv6 router isis 100
Device2(config-if-gigabitethernet0)# isis network point-to-point
Device2(config-if-gigabitethernet0)#exit
Device2(config)#interface gigabitethernet 1
Device2(config-if-gigabitethernet1)#ipv6 router isis 100
Device2(config-if-gigabitethernet1)# isis network point-to-point
Device2(config-if-gigabitethernet1)#exit
```

#Configure the IPv6 IS-IS process as 100, area number as 10, and type as Level-2 and enable the process on the interface of Device3. Configure the interface network type as point-to-point.

```
Device3#configure terminal
Device3(config)#router isis 100
Device3(config-isis)#net 10.0000.0000.0003.00
Device3(config-isis)#is-type level-2
Device3(config-isis)#metric-style wide
Device3(config-isis)#exit
Device3(config)#interface gigabitethernet0
Device3(config-if-gigabitethernet0)#ipv6 router isis 100
Device3(config-if-gigabitethernet0)# isis network point-to-point
Device3(config-if-gigabitethernet0)#exit
Device3(config)#interface gigabitethernet1
Device3(config-if-gigabitethernet1)#ipv6 router isis 100
Device3(config-if-gigabitethernet1)# isis network point-to-point
Device3(config-if-gigabitethernet1)#exit
Device3(config-if-gigabitethernet2)#ipv6 router isis 100
Device3(config-if-gigabitethernet2)# isis network point-to-point
Device3(config-if-gigabitethernet2)#exit
```

Step 3: On the interface gigabitethernet0 of Device3, configure the echo function of ipv6 bfd.

```
Device3(config)#interface gigabitethernet0
Device3(config-if-gigabitethernet0)# ipv6 bfd echo
Device3(config-if-gigabitethernet0)#exit
```

Step 4: Configure the routing policy.

#Configure Device1, configure route-map, and call ACL to only match 1001:2::1/64, while the other segments are filtered out.

```
Device1(config)#ipv6 access-list extended 7001
Device1(config-v6-list)#permit ipv6 1001:2::1/64 any
Device1(config-v6-list)#exit
Device1(config)#route-map ipv6frr_isis
Device1(config-route-map)#match ipv6 address 7001
Device1(config-route-map)#exit
```

Step 5: Configure the static fast re-routing.

```
Device1(config)#router isis 100
Device1(config-isis)#address-family ipv6 unicast
Device1(config-isis-af)# fast-reroute loop-free-alternate route-map ipv6frr_isis
Device1(config-isis-af)#exit-address-family
Device1(config-isis)#exit
```

Step 6: Check the result.

#View the Ipv6 IS-IS route table of Device1.

```
Device1#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management
L   ::1/128 [0/0]
    via ::, 06:22:27, lo0
C   1001:1::/64 [0/0]
    via ::, 06:17:34, gigabitethernet2
L   1001:1::1/128 [0/0]
    via ::, 06:17:34, gigabitethernet2
i   1001:2::/64 [115/20]
    via fe80::201:7aff:fe92:e6b6, 00:10:32, gigabitethernet0
C   2001:1::/64 [0/0]
    via ::, 01:24:08, gigabitethernet0
L   2001:1::1/128 [0/0]
    via ::, 01:24:08, gigabitethernet0
C   2001:2::/64 [0/0]
    via ::, 06:16:52, gigabitethernet1
L   2001:2::1/128 [0/0]
    via ::, 06:16:52, gigabitethernet1
i   2001:3::/64 [115/20]
```

```
    via fe80::201:7aff:fe92:e6b6, 00:29:05, gigabitethernet0
    via fe80::ced8:1fff:fe10:7aae, 00:35:19, gigabitethernet1
```

It can be seen from the routing table that route 1001:2::/64 preferably communicates with the line between Device1 and Device3.

#View the IPv6 FRR routing table of Device1.

```
Device1#show ipv6 frr route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
     U - Per-user Static route
     O - OSPF, OE-OSPF External, M - Management
i   1001:2::/64 [115/4294967295]
    via fe80::ced8:1fff:fe10:7aae, 00:07:18, gigabitethernet1
```

You can see that the next hop of the frr route 1001:2::/64 is link local address fe80::ced8:1fff:fe10:7aae, and the outgoing interface is gigabitethernet1.

#View the BFD session information of Device1.

```
Device1#show bfd session ipv6 detail
Total ipv6 session number: 1
OurAddr                         NeighAddr              LD/RD        State     Holddown
Interface
fe80::201:7aff:fe7d:1f1b           fe80::201:7aff:fe92:e6b6    1024/1024      UP        500
gigabitethernet0
Type:ipv6 direct  Mode:echo
Local Discriminator:369  Remote Discriminator:369
Local State:UP  Remote State:UP  Up for: 0h:15m:38s  Number of times UP:1
Send Interval:100ms  Detection time:500ms(100ms*5)
Local Diag:0  Demand mode:0  Poll bit:0
Registered modules:FIB_MGR
```

You can see that FIB_MGR is linked with BFD successfully, the session is established normally, and the session mode is echo.

#After the line between Device1 and Device3 fails, it can quickly detect and switch to Device2 for communication. Check the routing table of Device1. In the routing table, the outgoing interface to the destination network 1001:2::/64 has been switched to the backup interface gigabitethernet1.

```
Device1#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
     U - Per-user Static route
     O - OSPF, OE-OSPF External, M - Management
L   ::1/128 [0/0]
    via ::, 06:29:08, lo0
C   1001:1::/64 [0/0]
    via ::, 06:24:15, gigabitethernet2
```

```
L   1001:1::1/128 [0/0]
    via ::, 06:24:15, gigabitethernet2
i   1001:2::/64 [115/30]
    via fe80::ced8:1fff:fe10:7aae, 00:00:01, gigabitethernet1
i   2001:1::/64 [115/30]
    via fe80::ced8:1fff:fe10:7aae, 00:00:01, gigabitethernet1
C   2001:2::/64 [0/0]
    via ::, 06:23:33, gigabitethernet1
L   2001:2::1/128 [0/0]
    via ::, 06:23:33, gigabitethernet1
i   2001:3::/64 [115/20]
    via fe80::ced8:1fff:fe10:7aae, 00:42:00, gigabitethernet1
```

10. IRMP

10.1. Overview

The IRMP (Internal Routing Message Protocol), compatible with the Cisco EIGRP, is a dynamic routing protocol based on the distance vector. The IRMP uses the DUAL (Diffusing Update Algorithm) to calculate the route between multiple devices in parallel, ensuring no loop circuit and fast convergence. The IRMP overcomes the low convergence speed of distance vector-based routing protocol. Besides, it does not need the link status routing protocol to run the Dijkstra algorithm and the related databases do not consume huge CPU resources or memory.

The IRMP protocol applies to the medium- and small-size network and supports multiple autonomous systems. The autonomous systems can operate independently without interfering each other.

10.2. IRMP Function Configuration

Table 10-1 IRMP function list

Configuration Task	
Configure the IRMP basic function	Enable the IRMP protocol
Configure the IRMP route generation	Configure the IRMP redistribution
Configure the IRMP route control	Configure the IRMP route summary
	Configure the IRMP administrative distance
	Configure the IRMP load balancing
	Configure the IRMP route filtering
	Configure the IRMP metric offset
	Configure the IRMP route metric value
Configure the IRMP network optimization	Configure the IRMP passive interface
	Configure the IRMP stub mode
	Configure the IRMP intelligent query

Configuration Task	
Configure the IRMP network optimization	Configure the IRMP horizontal split
	Configure the IRMP timer
	Configure the IRMP static neighbor
Configure the IRMP network authentication	Configure the IRMP authentication

10.2.1. Configure IRMP Basic Function

In the various IRMP configuration tasks, the IRMP protocol must be enabled first to make other function feature configurations valid.

Configuration Condition

Before configuring the IRMP basic function, first complete the following tasks:

- Configure the link layer protocol to ensure the normal communication at the link layer.
- Configure the IP address of the interface to enable the neighboring nodes to be reachable at the network layer.

Enable IRMP Protocol

Before using the IRMP protocol, first the user must perform the following configurations:

- Establish the IRMP process.
- Configure the IRMP to cover the directly connected network segment.

Table 10-2 Enable the IRMP protocol

Step	Command	Description
Enter the global configuration mode	**configure terminal**	-
Establish the IRMP process and enter the IRMP configuration mode	**router irmp** *autonomous-system-number*	Mandatory By default, the IRMP process is not enabled.

Step	Command	Description
Configure the IRMP to cover the directly connected network segment	**network** *ip-address* [*wildcard-mask*]	Mandatory By default, no directly connected network is covered. If the wildcard mask is not carried, the classful network address is used by default.

Note:

- When establishing the IRMP neighbor, the autonomous system numbers must be consistent. Otherwise, the neighbor relationship cannot be established.

10.2.2. Configure IRMP Route Generation

In the IRMP, the **network** command can be used to cover the directly connected network segment routing. The external routing can be introduced using the redistribution.

Configuration Condition

Before configuring the IRMP route generation, first complete the following tasks:

- Configure the IP address of the interface to enable the neighboring nodes to be reachable at the network layer.
- Enable the IRMP protocol.

Configure the IRMP redistribution

Introduce the routing generated by other protocols to the IRMP through configuring the routing redistribution. The routing of other IRMP processes can also be introduced.

Table 10-3 Configure the IRMP redistribution

Step	Command	Description
Enter the global configuration mode	**configure terminal**	-
Enter the IRMP configuration mode	**router irmp** *autonomous-system-number*	-
Configure the IRMP redistribution	**redistribute** *protocol* [*process-id*] [**route-map** *route-map-name* / **metric** *bandwidth delay reliability loading mtu*]	Mandatory By default, the external routing is not redistributed.

Step	Command	Description
Configure the default metric value for IRMP redistributing external routing	**default-metric** *bandwidth delay reliability loading mtu*	Optional

Note:

- When the **default-metric** command and the **redistribute** *protocol* [*process-id*] **metric** command are configured at the same time, the later command has a higher priority.

10.2.3. Configure IRMP Route Control

Configuration Condition

Before configuring the IRMP route control, first complete the following tasks:

- Configure the IP address of the interface to enable the neighboring nodes to be reachable at the network layer.
- Enable the IRMP protocol.

Configure IRMP Route Summary

Both the IRMP aggregated route and original detailed route exist in the local route table, but only the aggregated routing is informed to the neighbor. Thus, in the large-size network, the neighboring route table scale reduces and the network bandwidth consumed by the protocol packets also reduces.

The IRMP supports the automatic summary and interface IP address summary. The metric value of the aggregated route uses the minimum metric value of all the detailed routes.

1. Automatic summary

In this mode, the route is aggregated as the corresponding classful network address by the non-manual configuration. The automatic route mode only aggregates the directly connected network segment route.

Table 10-4 Configure the IRMP automatic summary

Step	Command	Description
Enter the global configuration mode	**configure terminal**	-
Enter the IRMP configuration mode	**router irmp** *autonomous-system-number*	-

Step	Command	Description
Configure the IRMP automatic summary	**auto-summary**	Optional By default, the IRMP automatic summary function is disabled.

1. Interface route summary

The interface route summary needs the user to configure the combination of a pair of destination IP address and mask. This combination aggregates the routes that are covered in this network segment.

Table 10-5 Configure the IRMP interface route summary

Step	Command	Description
Enter the global configuration mode	**configure terminal**	-
Enter the interface configuration mode	**interface** *interface-name*	-
Configure the IRMP interface IP address summary	**ip summary-address irmp** *autonomous-system-number ip-address mask*	Mandatory

Note:

- The automatic summary only aggregates the directly connected network segment routes and nay cause all neighbors under the process to reestablish.
- The interface route summary aims at all the IRMP routes delivered from the specified interface and may cause neighbors on the specified interface to reestablish.
- The interface route summary priors to the automatic summary.
- The interface route summary is independent with each other. That is, when the interface route summary 10.1.0.0/16 and 10.0.0.0/8 are configured at the same time, the IRMP will advertise the two summary routes simultaneously.

Configure IRMP Administrative Distance

The administrative distance performs the routing policy for the routing in the same network augment from different protocols. The smaller the administrative distance is, the higher the priority of the routing is. You can change the administrative distance of the IRMP routing to affect the routing policy.

Table 10-6 Configure the IRMP administrative distance

Step	Command	Description
Enter the global configuration mode	**configure terminal**	-
Enter the IRMP configuration mode	**router irmp** *autonomous-system-number*	-
Configure the IRMP administrative distance	**distance** { **irmp** *internal-routes-distance external-routes-distance* | **summary** *summary-routes-distance* }	Optional By default, the administrative distance for the IRMP internal route is 90 and for the external route is 170. The management distance of the summary route is 5.

Note:

- When configuring the IRMP administrative distance, all neighbors under the process may be reestablished.

Configure IRMP Load Balancing

The IRMP supports the equal-cost load balancing and unequal-cost load balancing. When multiple routes with the same metric values to the destination network exist, equal-cost load balancing forms. When multiple routes with the different metric values to the destination network exist, you can run the **variance** command to change the conversion factor and form the unequal-cost load balancing.

Variance defines a conversion factor, which is used to determine the scope of the unequal-cost load balancing routing. When the metric value of a non-optimal routing is smaller than the metric value of the optimal routing multiplying the variance value, this routing is selected as the unequal-cost load balancing routing.

The **maximum-paths** command is used to control the maximum number of routings for IRMP load balancing, supporting a maximum of 6 paths for load balancing. When *path-number* is set to 1, the load function is cancelled. The number of paths for IRMP load balancing includes the number of equal-cost load balancing paths and unequal-cost load balancing paths. The path selected for IRMP load balancing depends on conversion factor variance.

Table 10-7 Configure the IRMP load balancing

Step	Command	Description
Enter the global configuration mode	**configure terminal**	-

Step	Command	Description
Enter the IRMP configuration mode	**router irmp** *autonomous-system-number*	-
Configure the IRM conversion factor to form the unequal load balancing	**variance** *number*	Optional By default, the conversion factor is 1.
Configure the maximum number of paths for IRMP load balancing	**maximum-paths** *path-number*	Optional By default, the maximum number of paths for IRMP load balancing is 4.

Configure IRMP Route Filtering

Control the received or delivered routing through configuring the ACL or prefix list. The inbound and outbound routing filtering can be configured on the interface at the same time.

Table 10-8 Configure the IRMP route filtering

Step	Command	Description
Enter the global configuration mode	**configure terminal**	-
Enter the IRMP configuration mode	**router irmp** *autonomous-system-number*	-
Configure the IRMP route filtering	**distribute-list** { *access-list-name* | **gateway** *prefix-list-name1* | **prefix** *prefix-list-name2* [**gateway** *prefix-list-name3*] } { **in** | **out** } [*interface-name*]	Mandatory

Note:

- The IRMP routing filtering only supports the standard ACL.
- You can configure the filtering on all interfaces or use the *interface-name* parameter to specify the interface for configuring filtering. When the preceding two modes are configured at the same time, the configuration is valid when the preceding two filtering rules are set to permit.

Configure IRMP Metric Offset

By default, the IRMP routing adopts the metric value advertised by the neighbor. In some application scenarios, the metric value requires modification. The user can rectify the metric value of a specified routing by configuring the IRMP metric offset.

If the inbound metric offset is configured, the routing metric value is modified, saved in the route table, and then advertised to the neighbor upon receiving the IRMP routing. If the outbound metric offset is configured, only the metric value advertised to the neighboring routing will be modified and the local metric value remains unchanged.

Table 10-9 Configure the IRMP metric offset

Step	Command	Description
Enter the global configuration mode	**configure terminal**	-
Enter the IRMP configuration mode	**router irmp** *autonomous-system-number*	-
Configure the IRMP metric offset	**offset-list** *access-list-name* { **in** | **out** } *offset-value* [*interface-name*]	Mandatory

Note:

- The IRMP metric offset only supports the standard ACL.
- When configuring the IRMP metric offset, all neighbors under the process may be reestablished.

Configure IRMP Route Metric Value

The IRMP calculates a composite metric value as the rout metric value based on the link features, such as link bandwidth, delay, load, and reliability. The calculation formula is as follows:

Route metric = 256 x ([k1 x (10^7 /Bandwidth) + k2 x (10^7 /Bandwidth)/(256 – Load) + k3 x (Delay/10)] x [k5/(Reliability + k4)])

Where, *bandwidth* is in the unit of kbit/s and *delay* is in the unit of microsecond. Each *k* value indicates the weighted value of the corresponding link feature. By default, the RMP uses the link bandwidth and delay as the metric value.

Table 10-10 Configure the IRMP route metric value

Step	Command	Description
Enter the global configuration mode	**configure terminal**	-

Step	Command	Description
Enter the IRMP configuration mode	**router irmp** *autonomous-system-number*	-
Configure the IRMP route metric value	**metric weights** *tos k1 k2 k3 k4 k5*	Mandatory *Tos* indicates the service type, supporting only the type-0 services. By default, *k1*=*k3*=1 and *k2*=*k4*=*k5*=0. The K value must be consistent when establishing the neighbor relationship.

10.2.4. Configure IRMP Network Optimization

Configuration Condition

Before configuring the IRMP network optimization, first complete the following tasks:

- Configure the IP address of the interface to enable the neighboring nodes to be reachable at the network layer.
- Enable the IRMP protocol.

Configure IRMP Passive Interface

The dynamic routing protocol adopts the passive interface to effectively reduce the network bandwidth consumed by the routing protocol. In the IRMP protocol, the passive interface is configured, which can suppress the IRMP packet receiving and transmitting on the specified interface. For example, if the gigabitethernet 0 is configured as the passive interface, the IRMP does not receive and transmit the packet on this interface even if the **network** command is used to cover the network segment that the interface locates at.

Table 10-11 Configure the IRMP passive interface

Step	Command	Description
Enter the global configuration mode	**configure terminal**	-
Enter the IRMP configuration mode	**router irmp** *autonomous-system-number*	-
Configure the IRMP passive interface	**passive interface** *interface-name*	Mandatory

Configure IRMP Stub Mode

Configuring the IRMP stub mode can effectively improve the IRMP network stability and reduce the resource occupation. The Stub function is usually used in the hub and spoke network topology. To enable the Stub function, you just need to configure Stub on the spoke router and do not need to change the configuration of the hub router. The spoke router specifies the advertised route type via the **irmp stub** command. The stub router does not send the query packet to the stub neighbor.

Table 10-12Configure the IRMP Stub

Step	Command	Description
Enter the global configuration mode	**configure terminal**	-
Enter the IRMP configuration mode	**router irmp** *autonomous-system-number*	-
Configure the IRMP Stub	**irmp stub** { **receive-only** } | { **connected** / **summary** / **redistributed** }	Mandatory By default, do not configure the stub mode.

Note:

- Modifying the stub configuration will result in the re-set up of all neighbors in the process.

Configure IRMP Intelligent Query

Configuring the IRMP intelligent query to control the sending of the query packet can effectively reduce the number of the packets interacted in the network and the resource occupation. After enabling the intelligent query function, the device does not send the query packet to the neighbor canceling the route, so as to reduce the number of the query packets and response packets in the network.

Table 10-13Configure the IRMP intelligent query

Step	Command	Description
Enter the global configuration mode	**configure terminal**	-
Enter the IRMP configuration mode	**router irmp** *autonomous-system-number*	-
Configure the IRMP intelligent query	**smart-query infinity-update**	Mandatory By default, do not enable the intelligent query.

Configure IRMP Horizontal Split

The IRMP horizontal split indicates that the route learned by the IRMP from a certain interface will not be advertised out from this interface to avoid forming a loop.

Table 10-14 Configure the IRMP horizontal split

Step	Command	Description
Enter the global configuration mode	**configure terminal**	-
Enter the interface configuration mode	**interface** *interface-name*	-
Configure the IRMP horizontal split	**ip split-horizon irmp** *autonomous-system-number*	Optional By default, the horizontal split function is enabled.

Configure IRMP Timer

1. Configure the keep-alive timer for the IRMP neighbor.

The Hello packet of the is used to discover the neighbor and keep it alive by spreading the packet on the network using the periodical and unreliable multicast mode with the multicast address as 224.0.0.10. the default delivery interval for the Hello packet is determined by the network type of the delivery interface. The default delivery interval is 5s on the broadcast and point-to-point interface and is 60s on the NBMA interface.

The hold time indicates the invalid interval of the IRMP neighbor. When a device receives a Hello packet form its neighbor, this packet contains a hold time, which informs the maximum valid time of the neighbor to the device. If the Hello packet is not received from the neighbor when times out, it indicates that the neighbor is not reachable and the neighbor will be removed. By default, the hold time is three times of the hello interval.

Table 10-15 Configure the keep-alive timer for the IRMP neighbor

Step	Command	Description
Enter the global configuration mode	**configure terminal**	-
Enter the interface configuration mode	**interface** *interface-name*	-

Step	Command	Description
Configure the IRMP hello interval	**ip hello-interval irmp** *autonomous-system hello-time*	Optional By default, the hello interval is set to 5s or 60s based on the network type.
Configure the IRMP hold time	**ip hold-time irmp** *autonomous-system-number hold-time*	Optional By default, the hold time is three times of the hello interval.

1. Configure the IRMP route timer.

The IRMP uses the DUAL (Diffusing Update Algorithm) to learn the routing. When the DUAL starts, the Active-timer is enabled. If the response of the queried packets is not received when the timer times out, the neighbors without response will be removed from the neighbor list. The route without response I set to Active and the route is considered as unreachable.

When the IRMP receives the neighbor request, if the local route is learnt from other neighbors, the IRMP will not respond immediately. Instead, enable a hold-down timer and then wait for a period. In the period, if the IRMP does not receive the request packet form the neighbor, it responds the packet to avoid forming the loop.

Table 10-16 Configure the IRMP route timer

Step	Command	Description
Enter the global configuration mode	**configure terminal**	-
Enter the IRMP configuration mode	**router irmp** *autonomous-system-number*	-
Configure the active-timer timeout	**timers active-time** *minutes*	Optional By default, the active-timer timeout is 3 minutes.
Configure the hold down timeout	**timers holddown-time** *seconds*	Optional By default, the hold down timeout is 5s.

Note:

- When configuring the route timer, all neighbors under the process may be reestablished.

Configure IRMP Static Neighbor

Configure the IRMP static neighbor, where all IRMP packets between neighbors adopt unicast interaction.

To run the IRMP dynamic routing protocol on the NBMA which does not support the broadcast network, such as X.25 and Frame, the static neighbor requires to be configured in pair. Otherwise the neighbor relationship will fail.

Table 10-17 Configure the IRMP static neighbor

Step	Command	Description
Enter the global configuration mode	**configure terminal**	-
Enter the IRMP configuration mode	**router irmp** *autonomous-system-number*	-
Configure the IRMP static neighbor	**neighbor** *neighbor-ip-address interface-name*	Mandatory

10.2.5. Configure IRMP Network Authentication

Configuration Condition

Before configuring the IRMP authentication, first complete the following tasks:

- Configure the IP address of the interface to enable the neighboring nodes to be reachable at the network layer.
- Enable the IRMP protocol.

Configure IRMP Authentication

Perform validity check and verification on the IRMP packet by configuring the IRMP authentication to improve the network security. The IRMP only supports the MD5 authentication.

Table 10-18 Configure the IRMP authentication

Step	Command	Description
Enter the global configuration mode	**configure terminal**	-
Enter the interface configuration mode	**interface** *interface-name*	-

Step	Command	Description
Configure the IRMP authentication function	**ip message-digest-key irmp** *autonomous-system-number key-id* **md5** { **0** | **7** } *password*	Mandatory

Note:

- When the same Key-id and password are configured on the interfaces at the same time, the IRMP neighboring relationship can be established.

10.2.6. IRMP Monitoring and Maintaining

Table 10-19 The IRMP monitoring and maintaining

Command	Description
clear ip irmp error	Clear the statistics information of the IRMP error packet
clear ip irmp [*autonomous-system-number*] **neighbors** [*interface-name*]	Reset the IRMP neighbor
clear ip irmp traffic [*autonomous-system-number*]	Clear the statistics information for receiving and transmitting the IRMP packet
show i+p irmp error	Display the statistics information of the IRMP error packet
show ip irmp interface [*autonomous-system-number*] | [*interface-name*]	Display the IRMP interface information
show ip irmp neighbor [*autonomous-system-number* **| detail |** *interface-name*]	Display the IRMP neighbor information
show ip irmp topology [*autonomous-system-number*] [**active | detail | summary** | *ip-address mask*]	Display the routing information in the IRMP topology table
show ip irmp traffic [*autonomous-system-number*]	Display the statistics information of receiving and transmitting the IRMP packet

10.3. IRMP Typical Configuration Example

10.3.1. Configure IRMP Basic Function

Network Requirements

- The IRMP operates between Device1 and Device2, establishing the neighbor and interacting the route.

Network Topology

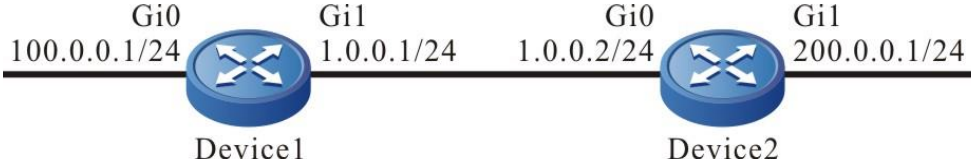

Figure 10–1 Networking of the IRMP basic function

Configuration Steps

Step 1: Configure the IP address of the interfaces. (Omitted)

Step 2: Configure the IRMP.

#Configure Device1.

```
Device1#configure terminal
Device1(config)#router irmp 100
Device1(config-irmp)#network 1.0.0.0 0.0.0.255
Device1(config-irmp)#network 100.0.0.0 0.0.0.255
Device1(config-irmp)#exit
```

#Configure Device2.

```
Device2#configure terminal
Device2(config)#router irmp 100
Device2(config-irmp)#network 1.0.0.0 0.0.0.255
Device2(config-irmp)#network 200.0.0.0 0.0.0.255
Device2(config-irmp)#exit
```

Step 3: Check the result.

#View the IRMP neighbor information of Device1.

```
Device1#show ip irmp neighbor
IP-IRMP neighbors for process 100 Total neighbor 1
Address      Interface              Hold(s)  Uptime   SeqNum   Srtt(ms)  Rto(s)
1.0.0.2    gigabitethernet1       14     00:00:47  7      0      2
```

#View the IRMP neighbor information of Device2.

```
Device2#show ip irmp neighbor
IP-IRMP neighbors for process 100 Total neighbor 1
Address      Interface              Hold(s)  Uptime   SeqNum   Srtt(ms)  Rto(s)
```

```
1.0.0.1          gigabitethernet0          11       00:00:43  4       3920      4
```

It can be viewed that the IRMP neighbor is established between Device1 and Device2l

#View the topology table and route table of Device1.

```
Device1#show ip irmp topology
IP-IRMP Topology Table for process 100
Codes:  P - Passive, A - Active, H - Holddown, D - Hidden
        > - FIB route, * - FIB successor

P >1.0.0.0/24, 1 successors, FD is 2816
        *via Connected (2816/0), gigabitethernet1
P >200.0.0.0/24, 1 successors, FD is 3072
        *via 1.0.0.2 (3072/2816), gigabitethernet1
P >100.0.0.0/24, 1 successors, FD is 2816
        *via Connected (2816/0), gigabitethernet0

Device1#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

C   1.0.0.0/24 is directly connected, 00:27:02, gigabitethernet1
L   1.0.0.1/32 is directly connected, 00:27:02, gigabitethernet1
E   200.0.0.0/24 [90/3072] via 1.0.0.2, 00:15:07, gigabitethernet1
C   100.0.0.0/24 is directly connected, 00:27:14, gigabitethernet0
L   100.0.0.1/32 is directly connected, 00:27:14, gigabitethernet0
C   127.0.0.0/8 is directly connected, 137:46:13, lo0
L   127.0.0.1/32 is directly connected, 137:46:13, lo0
```

Device1 learns the routing 200.0.0.0/24 advertised by Device2.

#View the IRMP topology table and route table of Device2.

```
Device2#show ip irmp topology
IP-IRMP Topology Table for process 100
Codes:  P - Passive, A - Active, H - Holddown, D - Hidden
        > - FIB route, * - FIB successor

P >1.0.0.0/24, 1 successors, FD is 2816
        *via Connected (2816/0), gigabitethernet0
P >200.0.0.0/24, 1 successors, FD is 2816
```

```
        *via Connected (2816/0), gigabitethernet1
P >100.0.0.0/24, 1 successors, FD is 3072
        *via 1.0.0.1 (3072/2816), gigabitethernet0

Device2#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

C   1.0.0.0/24 is directly connected, 00:20:12, gigabitethernet0
L   1.0.0.2/32 is directly connected, 00:20:12, gigabitethernet0
C   200.0.0.0/24 is directly connected, 00:19:53, gigabitethernet1
L   200.0.0.1/32 is directly connected, 00:19:53, gigabitethernet1
E   100.0.0.0/24 [90/3072] via 1.0.0.1, 00:08:41, gigabitethernet0
C   127.0.0.0/8 is directly connected, 361:04:08, lo0
L   127.0.0.1/32 is directly connected, 361:04:08, lo0
```

Device2 learns the routing 100.0.0.0/24 advertised by Device1.

10.3.2. Configure IRMP Redistribution

Network Requirements

- The OSPF neighbor is established between Device3 and Device2 and the interface directly connected routing 200.0.0.0/24 and 210.0.0.0/24 are advertised to Device2.
- The IRMP neighbor is established between Device1 and Device2. When Device2 redistributing the OSPF routing to the IRMP, only the routing 200.0.0.0/24 instead of the routing 210.0.0.0/24 is advertised to Device1 through the routing policy control.

Network Topology

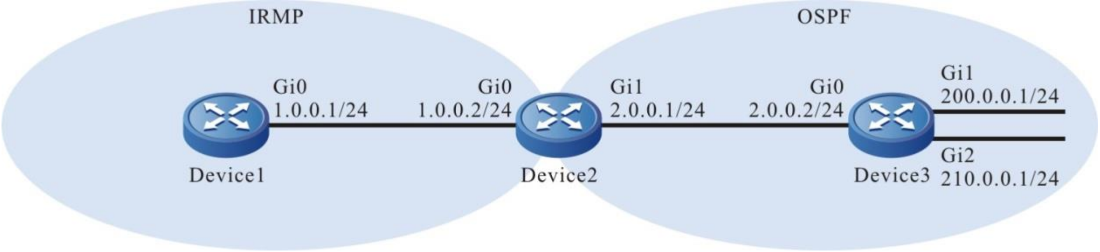

Figure 10–2 Networking of the IRMP route redistribution

Configuration Steps

Step 1: Configure the IP address of the interfaces. (Omitted)

Step 2: Configure the OSPF.

#Configure Device2.

```
Device2#configure terminal
Device2(config)#router ospf 100
```

```
Device2(config-ospf)#network 2.0.0.0 0.0.0.255 area 0
Device2(config-ospf)#exit
```

#Configure Device3.

```
Device3#configure terminal
Device3(config)#router ospf 100
Device3(config-ospf)#network 2.0.0.0 0.0.0.255 area 0
Device3(config-ospf)#network 200.0.0.0 0.0.0.255 area 0
Device3(config-ospf)#network 210.0.0.0 0.0.0.255 area 0
Device3(config-ospf)#exit
```

#View the route table of Device2.

```
Device2#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
     U - Per-user Static route
     O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

C   1.0.0.0/24 is directly connected, 07:39:21, gigabitethernet0
L   1.0.0.2/32 is directly connected, 07:39:21, gigabitethernet0
C   2.0.0.0/24 is directly connected, 00:03:36, gigabitethernet1
L   2.0.0.1/32 is directly connected, 00:03:36, gigabitethernet1
C   127.0.0.0/8 is directly connected, 57:04:58, lo0
L   127.0.0.1/32 is directly connected, 57:04:58, lo0
O   200.0.0.0/24 [110/2] via 2.0.0.2, 00:01:10, gigabitethernet1
O   210.0.0.0/24 [110/2] via 2.0.0.2, 00:01:10, gigabitethernet1
```

In the route table, it can be viewed that Device2 learns the OSPF routing 200.0.0.0/24 and 210.0.0.0/24 advertised by Device3.

Step 3: Configure the IRMP.

#Configure Device1.

```
Device1#configure terminal
Device1(config)#router irmp 100
Device1(config-irmp)#network 1.0.0.0 0.0.0.255
Device1(config-irmp)#exit
```

#Configure Device2.

```
Device2(config)#router irmp 100
Device2(config-irmp)#network 1.0.0.0 0.0.0.255
Device2(config-irmp)#exit
```

#View the IRMP neighbor informatione of Device1.

```
Device1#show ip irmp neighbor
```

```
IP-IRMP neighbors for process 100 Total neighbor 1
Address        Interface              Hold(s)  Uptime   SeqNum   Srtt(ms)  Rto(s)
1.0.0.2      gigabitethernet0        12      00:01:03  1       0       2
```

The IRMP neighbor is successfully established between Device1 and Device2.

Step 4: Configure the IRMP to redistribute the OSPF routing and to coordinate with the routing policy.

#Configure the routing policy to match the ACL only permitting the routing 200.0.0.0/24 on Device2 and to coordinate with the routing policy when the IRMP redistributing the OSPF routing.

```
Device2(config)#ip access-list standard 1
Device2(config-std-nacl)#permit 200.0.0.0 0.0.0.255
Device2(config-std-nacl)#exit
Device2(config)#route-map ospf_to_irmp
Device2(config-route-map)#match ip address 1
Device2(config-route-map)#exit
Device2(config)#router irmp 100
Device2(config-irmp)#redistribute ospf 100 route-map ospf_to_irmp
Device2(config-irmp)#exit
```

Note:

- When configuring the routing policy, both the prefix list and ACL can establish the matching rule. The difference lies in that the prefix list can accurately match the route mask, but the ACL cannot match the route mask.

Step 5: Check the result.

#View the IRMP topology table of Device2.

```
Device2#show ip irmp topology
IP-IRMP Topology Table for process 100
Codes:  P - Passive, A - Active, H - Holddown, D - Hidden
     > - FIB route, * - FIB successor

P >1.0.0.0/24, 1 successors, FD is 2816
     *via Connected (2816/0), gigabitethernet0
P  200.0.0.0/24, 1 successors, FD is 2816
      via RedisOSPF 100 (2816/0)
```

It can be viewed that Device2 successfully distributes the OSPF routing to the IRMP.

#View the topology table and route table of Device1.

```
Device1#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
     U - Per-user Static route
```

```
          O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

C   1.0.0.0/24 is directly connected, 07:47:01, gigabitethernet0
L   1.0.0.1/32 is directly connected, 07:47:01, gigabitethernet0
C   127.0.0.0/8 is directly connected, 07:55:23, lo0
L   127.0.0.1/32 is directly connected, 07:55:23, lo0
Ex  200.0.0.0/24 [170/3072] via 1.0.0.2, 00:00:06, gigabitethernet0
```

It can be viewed that Device1 learns the routing 200.0.0.0/24.

Caution:

- In the actual application, if there are two or more edge routers in the autonomous system, you are advised to not redistribute routes between different routing protocols. If necessary, configure the routing control policies such as filtering and summary on the edge router in the autonomous system to prevent generating routing loop.

10.3.3. Configure IRMP Metric Offset

Network Requirements

- The IRMP protocol operates among Device1, Device2, Device3, and Device4 for interconnection.
- Device1 learns the routing 200.0.0.0/24 from Device2 and Device3 at the same time.
- Configure the routing metric offset at the receiving direction on Device1to enable Device1 to choose the routing advertised by Device2 preferentially.

Network Topology

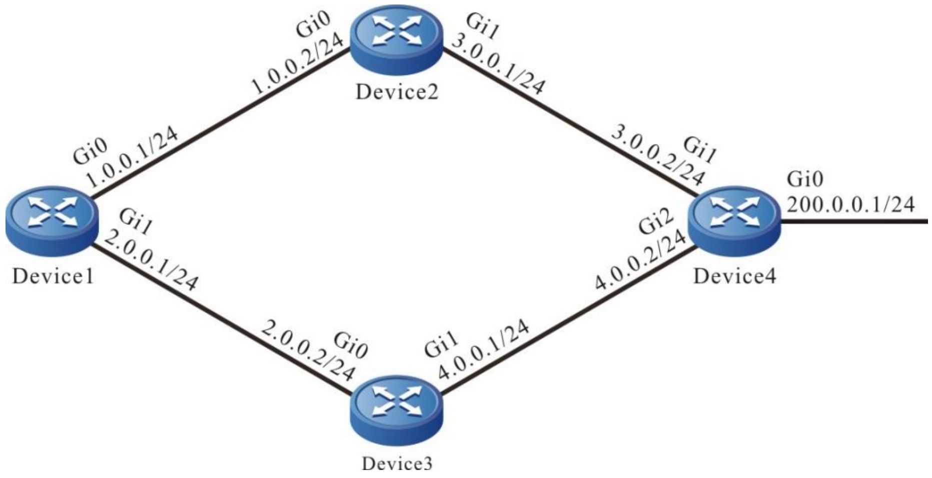

Figure 10–3 Networking of the IRMP metric offset

Configuration Steps

Step 1: Configure the IP address of the interfaces. (Omitted)

Step 2: Configure the IRMP.

#Configure Device1.

```
Device1#configure terminal
Device1(config)#router irmp 100
```

```
Device1(config-irmp)#network 1.0.0.0 0.0.0.255
Device1(config-irmp)#network 2.0.0.0 0.0.0.255
Device1(config-irmp)#exit
```

# Configure Device2.

```
Device2#configure terminal
Device2(config)#router irmp 100
Device2(config-irmp)#network 1.0.0.0 0.0.0.255
Device2(config-irmp)#network 3.0.0.0 0.0.0.255
Device2(config-irmp)#exit
```

# Configure Device3.

```
Device3#configure terminal
Device3(config)#router irmp 100
Device3(config-irmp)#network 2.0.0.0 0.0.0.255
Device3(config-irmp)#network 4.0.0.0 0.0.0.255
Device3(config-irmp)#exit
```

# Configure Device4.

```
Device4#configure terminal
Device4(config)#router irmp 100
Device4(config-irmp)#network 3.0.0.0 0.0.0.255
Device4(config-irmp)#network 4.0.0.0 0.0.0.255
Device4(config-irmp)#network 200.0.0.0 0.0.0.255
Device4(config-irmp)#exit
```

#View the IRMP neighbor information of Device1.

```
Device1#show ip irmp neighbor
IP-IRMP neighbors for process 100 Total neighbor 2
Address        Interface                Hold(s)   Uptime    SeqNum    Srtt(ms)  Rto(s)


1.0.0.2       gigabitethernet0         11       00:10:37  10       0        2
2.0.0.2        gigabitethernet1         12       00:10:15  9       0        2
```

#View the IRMP neighbor information of Device4.

```
Device4#show ip irmp neighbor
IP-IRMP neighbors for process 100 Total neighbor 2
Address        Interface                Hold(s)   Uptime    SeqNum    Srtt(ms)  Rto(s)
3.0.0.1       gigabitethernet1         14       00:11:37  13       0        2
4.0.0.1       gigabitethernet2         12       00:10:45  12       0        2
```

Device1 sets up the IRMP neighbor with Device2, Device3 respectively. Device4 sets up the IRMP neighbor with Device2, Device3 respectively.

#View the topology table and route table of Device1.

```
Device1#show ip irmp topology
IP-IRMP Topology Table for process 100
Codes:  P - Passive, A - Active, H - Holddown, D - Hidden
        > - FIB route, * - FIB successor

P >1.0.0.0/24, 1 successors, FD is 2816
        *via Connected (2816/0), gigabitethernet0
/2/0P >2.0.0.0/24, 1 successors, FD is 2816
        *via Connected (2816/0), gigabitethernet1
P >3.0.0.0/24, 1 successors, FD is 3072
        *via 1.0.0.2 (3072/2816), gigabitethernet0
P >4.0.0.0/24, 1 successors, FD is 3072
        *via 2.0.0.2 (3072/512), gigabitethernet1
P >200.0.0.0/24, 2 successors, FD is 3328
        *via 2.0.0.2 (3328/3072), gigabitethernet1
        *via 1.0.0.2 (3328/3072), gigabitethernet0

Device1#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

C   1.0.0.0/24 is directly connected, 13:16:35, gigabitethernet0
L   1.0.0.1/32 is directly connected, 13:16:35, gigabitethernet0
C   2.0.0.0/24 is directly connected, 13:19:24, gigabitethernet1
L   2.0.0.1/32 is directly connected, 13:19:24, gigabitethernet1
E   3.0.0.0/24 [90/3072] via 1.0.0.2, 00:03:22, gigabitethernet0
E   4.0.0.0/24 [90/3072] via 2.0.0.2, 00:22:01, gigabitethernet1
C   127.0.0.0/8 is directly connected, 22:27:25, lo0
L   127.0.0.1/32 is directly connected, 22:27:25, lo0
E   200.0.0.0/24 [90/3328] via 2.0.0.2, 00:21:22, gigabitethernet1
                 [90/3328] via 1.0.0.2, 00:03:22, gigabitethernet0
```

There are two load balancing routing to the network segment 200.0.0.0/24 in Device1 route table. The forwarding paths to the network segment are Device1→Device2→Device4 and Device1→Device3→Device4.

Step 3: Configure the IRMP metric offset.

#Configure the offset list on Device1. Add metric value 100 to the interface connected to Device3 by the specified routing to enable the total metric value going through Device3 is larger than that going through Device2.

```
Device1(config)#ip access-list standard 1
Device1(config-std-nacl)#permit 200.0.0.0 0.0.0.255
Device1(config-std-nacl)#exit
Device1(config)#router irmp 100
Device1(config-irmp)#offset-list 1 in 100 gigabitethernet1
Device1(config-irmp)#exit
```

Step 4: Check the result.

#View the topology table and route table of Device1.

```
Device1#show ip irmp topology
IP-IRMP Topology Table for process 100
Codes:  P - Passive, A - Active, H - Holddown, D - Hidden
       > - FIB route, * - FIB successor

P >1.0.0.0/24, 1 successors, FD is 2816
       *via Connected (2816/0), gigabitethernet0
P >2.0.0.0/24, 1 successors, FD is 2816
       *via Connected (2816/0), gigabitethernet1
P >3.0.0.0/24, 1 successors, FD is 3072
       *via 1.0.0.2 (3072/2816), gigabitethernet0
P >4.0.0.0/24, 1 successors, FD is 3072
       *via 2.0.0.2 (3072/512), gigabitethernet1
P >200.0.0.0/24, 2 successors, FD is 3328
       *via 1.0.0.2 (3328/3072), gigabitethernet0
        via 2.0.0.2 (3428/3172), gigabitethernet1

Device1#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
      U - Per-user Static route
      O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

C   1.0.0.0/24 is directly connected, 13:33:22, gigabitethernet0
L   1.0.0.1/32 is directly connected, 13:33:22, gigabitethernet0
C   2.0.0.0/24 is directly connected, 13:36:11, gigabitethernet1
L   2.0.0.1/32 is directly connected, 13:36:11, gigabitethernet1
```

```
E   3.0.0.0/24 [90/3072] via 1.0.0.2, 00:06:56, gigabitethernet0
E   4.0.0.0/24 [90/3072] via 2.0.0.2, 00:05:43, gigabitethernet1
C   127.0.0.0/8 is directly connected, 22:44:12, lo0
L   127.0.0.1/32 is directly connected, 22:44:12, lo0
E   200.0.0.0/24 [90/3328] via 1.0.0.2, 00:06:56, gigabitethernet0
```

After the metric offset is configured, Device1 chooses the routing 200.0.0.0/24 advertised by Device2.

Note:

- Configuring the IRMP offset list may cause the neighbor to be reestablished.

10.3.4. Configure IRMP Route Filtering

Network Requirements

- The IRMP operated between Device1 and Device2 for routing interaction.
- Device1 learns two routing 2.0.0.0/24 and 3.0.0.0/24 advertised by Device2 and Device1 only reserves the routing information of 2.0.0.0/24.

Network Topology

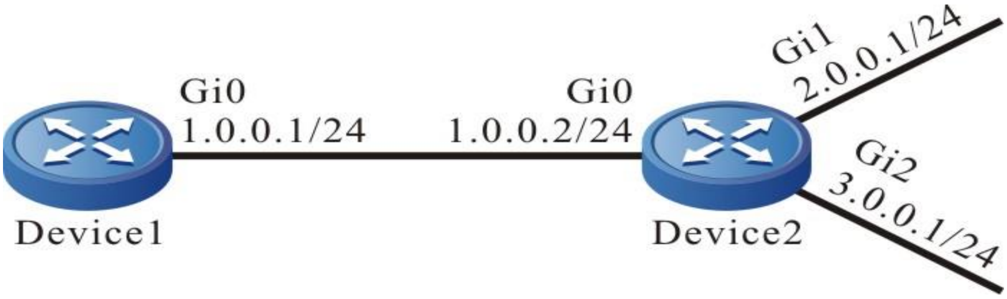

Figure 10–4 Networking of the IRMP route filtering

Configuration Steps

Step 1: Configure the IP address of the interfaces. (Omitted)

Step 2: Configure the IRMP.

#Configure Device1.

```
Device1#configure terminal
Device1(config)#router irmp 100
Device1(config-irmp)#network 1.0.0.0 0.0.0.255
Device1(config-irmp)#exit
```

#Configure Device2.

```
Device2#configure terminal
Device2(config)#router irmp 100
Device2(config-irmp)#network 1.0.0.0 0.0.0.255
Device2(config-irmp)#network 2.0.0.0 0.0.0.255
Device2(config-irmp)#network 3.0.0.0 0.0.0.255
Device2(config-irmp)#exit
```

#View the IRMP neighbor inofrmation of Device1.

```
Device1#show ip irmp neighbor
IP-IRMP neighbors for process 100 Total neighbor 1
Address         Interface                   Hold(s)  Uptime   SeqNum   Srtt(ms)  Rto(s)
1.0.0.2        gigabitethernet0          11       00:05:00  3        0        2
```

It can be viewed that the IRMP neighbor is successfully established between Device1 and Device2.

#View the route table of Device1.

```
Device1#show ip irmp topology
IP-IRMP Topology Table for process 100
Codes:  P - Passive, A - Active, H - Holddown, D - Hidden
        > - FIB route, * - FIB successor

P >1.0.0.0/24, 1 successors, FD is 2816
        *via Connected (2816/0), gigabitethernet0
P >2.0.0.0/24, 1 successors, FD is 3072
        *via 1.0.0.2 (3072/2816), gigabitethernet0
P >3.0.0.0/24, 1 successors, FD is 3072
        *via 1.0.0.2 (3072/2816), gigabitethernet0

Device1#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

C   1.0.0.0/24 is directly connected, 00:11:29, gigabitethernet0
L   1.0.0.1/32 is directly connected, 00:11:29, gigabitethernet0
E   2.0.0.0/24 [90/3072] via 1.0.0.2, 00:07:43, gigabitethernet0
E   3.0.0.0/24 [90/3072] via 1.0.0.2, 00:07:40, gigabitethernet0
C   127.0.0.0/8 is directly connected, 00:19:50, lo0
L   127.0.0.1/32 is directly connected, 00:19:50, lo0
```

Device1 learns the routing 2.0.0.0/24 and 3.0.0.0/24.

Step 3: Configure the route filtering.

#Configure the ACL rule to only permit the routing 2.0.0.0/24 on Device1 and configure the inbound filtering list to coordinate with the ACL in the IRMP.

```
Device1(config)#ip access-list standard 1
Device1(config-std-nacl)#permit 2.0.0.0 0.0.0.255
```

```
Device1(config-std-nacl)#exit
Device1(config)#router irmp 100
Device1(config-irmp)#distribute-list 1 in
Device1(config-irmp)#exit
```

Note:

- When configuring the route filtering, the prefix list and ACL can establish the matching rule. The difference lies in that the prefix list can accurately match the route mask, but the ACL cannot match the route mask.

Step 4: Check the result.

#View the topology table and route table of Device1.

```
Device1#show ip irmp topology
IP-IRMP Topology Table for process 100
Codes:  P - Passive, A - Active, H - Holddown, D - Hidden
        > - FIB route, * - FIB successor

P >1.0.0.0/24, 1 successors, FD is 2816
        *via Connected (2816/0), gigabitethernet0
P >2.0.0.0/24, 1 successors, FD is 3072
        *via 1.0.0.2 (3072/2816), gigabitethernet0

Device1#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

C   1.0.0.0/24 is directly connected, 00:22:33, gigabitethernet0
L   1.0.0.1/32 is directly connected, 00:22:33, gigabitethernet0
E   2.0.0.0/24 [90/3072] via 1.0.0.2, 00:04:18, gigabitethernet0
C   127.0.0.0/8 is directly connected, 00:30:55, lo0
L   127.0.0.1/32 is directly connected, 00:30:55, lo0
```

Device1 only learns the routing 2.0.0.0/24, but the routing 3.0.0.0/24 is successfully filtered.

Note:

- Configuring the **distribute-list** command may cause the IRMP neighbor to be reestablished.

10.3.5. Configure IRMP Route Summary

Network Requirements

- The IRMP protocol operates among Device1, Device2, Device3, and Device4 for routing interaction,
- Device1 learns two routing 100.1.0.0/24 and 100.2.0.0/24 from Device. To reduce the route table scale of Device1, it is required that Device2 only releases the route summary 100.0.0.0/14 to Device1.

Network Topology

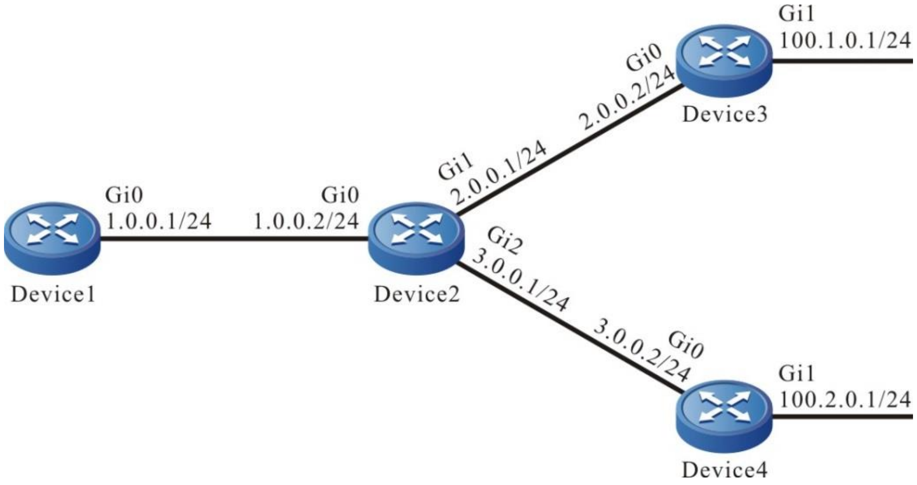

Figure 10–5 Networking of the IRMP route summary

Configuration Steps

Step 1: Configure the IP address of the interfaces. (Omitted)

Step 2: Configure the IRMP.

#Configure Device1.

```
Device1#configure terminal
Device1(config)#router irmp 100
Device1(config-irmp)#network 1.0.0.0 0.0.0.255
Device1(config-irmp)#exit
```

#Configure Device2.

```
Device2#configure terminal
Device2(config)#router irmp 100
Device2(config-irmp)#network 1.0.0.0 0.0.0.255
Device2(config-irmp)#network 2.0.0.0 0.0.0.255
Device2(config-irmp)#network 3.0.0.0 0.0.0.255
Device2(config-irmp)#exit
```

#Configure Device3.

```
Device3#configure terminal
Device3(config)#router irmp 100
```

```
Device3(config-irmp)#network 2.0.0.0 0.0.0.255
Device3(config-irmp)#network 100.1.0.0 0.0.0.255
Device3(config-irmp)#exit
```

#Configure Device4.

```
Device4#configure terminal
Device4(config)#router irmp 100
Device4(config-irmp)#network 3.0.0.0 0.0.0.255
Device4(config-irmp)#network 100.2.0.0 0.0.0.255
Device4(config-irmp)#exit
```

#View the IRMP neighbor inofrmation of Device2.

```
Device2#show ip irmp neighbor
IP-IRMP neighbors for process 100 Total neighbor 3
Address         Interface                   Hold(s)  Uptime   SeqNum   Srtt(ms)  Rto(s)
1.0.0.1        gigabitethernet0          11       00:05:04  4       0        2
2.0.0.2         gigabitethernet0          14       00:04:42  3       16       2
3.0.0.2         gigabitethernet0          11       00:04:00  2       0        2
```

Device2 successfully establishes the IRMP neighbor with Device1, Device3, and Device4, respectively.

#View the topology table and route table of Device2.

```
Device2#show ip irmp topology
IP-IRMP Topology Table for process 100
Codes:  P - Passive, A - Active, H - Holddown, D - Hidden
        > - FIB route, * - FIB successor

P >1.0.0.0/24, 1 successors, FD is 2816
        *via Connected (2816/0), gigabitethernet0
P >2.0.0.0/24, 1 successors, FD is 2816
        *via Connected (2816/0), gigabitethernet1
P >3.0.0.0/24, 1 successors, FD is 2816
        *via Connected (2816/0), gigabitethernet2
P >100.1.0.0/24, 1 successors, FD is 3072
        *via 2.0.0.2 (3072/2816), gigabitethernet1
P >100.2.0.0/24, 1 successors, FD is 3072
        *via 3.0.0.2 (3072/2816), gigabitethernet2

Device2#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
```

```
    U - Per-user Static route
    O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

C   1.0.0.0/24 is directly connected, 08:12:36, gigabitethernet0
L   1.0.0.2/32 is directly connected, 08:12:36, gigabitethernet0
C   2.0.0.0/24 is directly connected, 00:36:51, gigabitethernet1
L   2.0.0.1/32 is directly connected, 00:36:51, gigabitethernet1
C   3.0.0.0/24 is directly connected, 00:12:06, gigabitethernet2
L   3.0.0.1/32 is directly connected, 00:12:06, gigabitethernet2
E   100.1.0.0/24 [90/3072] via 2.0.0.2, 00:10:03, gigabitethernet1
E   100.2.0.0/24 [90/3072] via 3.0.0.2, 00:00:08, gigabitethernet2
C   127.0.0.0/8 is directly connected, 57:38:13, lo0
L   127.0.0.1/32 is directly connected, 57:38:13, lo0
```

#View the topology table and route table of Device1.

```
Device1#show ip irmp topology
IP-IRMP Topology Table for process 100
Codes:  P - Passive, A - Active, H - Holddown, D - Hidden
        > - FIB route, * - FIB successor

P >1.0.0.0/24, 1 successors, FD is 2816
        *via Connected (2816/0), gigabitethernet0
P >2.0.0.0/24, 1 successors, FD is 3072
        *via 1.0.0.2 (3072/2816), gigabitethernet0
P >3.0.0.0/24, 1 successors, FD is 3072
        *via 1.0.0.2 (3072/2816), gigabitethernet0
P >100.1.0.0/24, 1 successors, FD is 3328
        *via 1.0.0.2 (3328/3072), gigabitethernet0
P >100.2.0.0/24, 1 successors, FD is 3328
        *via 1.0.0.2 (3328/3072), gigabitethernet0

Device1#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
    U - Per-user Static route
    O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

C   1.0.0.0/24 is directly connected, 08:09:15, gigabitethernet0
L   1.0.0.1/32 is directly connected, 08:09:15, gigabitethernet0
```

```
E   2.0.0.0/24 [90/3072] via 1.0.0.2, 00:13:19, gigabitethernet0
E   3.0.0.0/24 [90/3072] via 1.0.0.2, 00:13:17, gigabitethernet0
E   100.1.0.0/24 [90/3328] via 1.0.0.2, 00:12:53, gigabitethernet0
E   100.2.0.0/24 [90/3328] via 1.0.0.2, 00:02:57, gigabitethernet0
C   127.0.0.0/8 is directly connected, 08:17:36, lo0
L   127.0.0.1/32 is directly connected, 08:17:36, lo0
```

Both Device1 and Device2 learn routing 100.1.0.0/24 and 100.2.0.0/24.

Step 3: Configure the IRMP route summary.

#Configure Device2 and configure the IRMP route summary 100.0.0.0/14 on the interface connected to Device1.

```
Device2(config)#interface gigabitethernet0
Device2(config-if-gigabitethernet0)#ip summary-address irmp 100 100.0.0.0 255.252.0.0
Device2(config-if-gigabitethernet0)#exit
```

Step 4: Check the result.

#View the IRMP topology table of Device2.

```
Device2#show ip irmp topology
IP-IRMP Topology Table for process 100
Codes:  P - Passive, A - Active, H - Holddown, D - Hidden
        > - FIB route, * - FIB successor


P >1.0.0.0/24, 1 successors, FD is 2816
        *via Connected (2816/0), gigabitethernet0
P >2.0.0.0/24, 1 successors, FD is 2816
        *via Connected (2816/0), gigabitethernet1
P >3.0.0.0/24, 1 successors, FD is 2816
        *via Connected (2816/0), gigabitethernet2
P  100.0.0.0/14, 1 successors, FD is 3072
         via AddrSumm (3072/0)
P >100.1.0.0/24, 1 successors, FD is 3072
        *via 2.0.0.2 (3072/2816), gigabitethernet1
P >100.2.0.0/24, 1 successors, FD is 3072
        *via 3.0.0.2 (3072/2816), gigabitethernet2
```

A route summary 100.0.0.0/14 is generated on Device2.

#View the topology table and route table of Device1.

```
Device1#show ip irmp topology
```

```
IP-IRMP Topology Table for process 100
Codes:  P - Passive, A - Active, H - Holddown, D - Hidden
        > - FIB route, * - FIB successor

P >1.0.0.0/24, 1 successors, FD is 2816
        *via Connected (2816/0), gigabitethernet0
P >2.0.0.0/24, 1 successors, FD is 3072
        *via 1.0.0.2 (3072/2816), gigabitethernet0
P >3.0.0.0/24, 1 successors, FD is 3072
        *via 1.0.0.2 (3072/2816), gigabitethernet0
P >100.0.0.0/14, 1 successors, FD is 3328
        *via 1.0.0.2 (3328/3072), gigabitethernet0

Device1#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

C   1.0.0.0/24 is directly connected, 08:18:49, gigabitethernet0
L   1.0.0.1/32 is directly connected, 08:18:49, gigabitethernet0
E   2.0.0.0/24 [90/3072] via 1.0.0.2, 00:04:54, gigabitethernet0
E   3.0.0.0/24 [90/3072] via 1.0.0.2, 00:04:54, gigabitethernet0
E   100.0.0.0/14 [90/3328] via 1.0.0.2, 00:04:54, gigabitethernet0
C   127.0.0.0/8 is directly connected, 08:27:10, lo0
L   127.0.0.1/32 is directly connected, 08:27:10, lo0
```

It can be viewed that only the route summary100.0.0.0/14 instead of the corresponding detailed route100.1.0.0/24 and 100.2.0.0/24 exists on Device1.

Note:

- Configuring the IRMP route summary in the interface mode may cause the neighbor to be reestablished.

10.3.6. Configure IRMP Load Balancing

Network Requirements

- The IRMP protocol operates among Device1, Device2, Device3, and Device4 for routing interaction.
- Device1 learns the routing 200.0.0.0/24 from Device2 and Device3 at the same time. Modify the bandwidth of interface gigabitethernet1 on Device1 to enable Device1 to preferentially choose the routing 200.0.0.0/24 learned from Device2.
- Device1 is required to transfer data to the network segment200.0.0.0/24 simultaneously on lines Device1→Device2→Device4 and Device1→Device3→Device4.

Network Topology

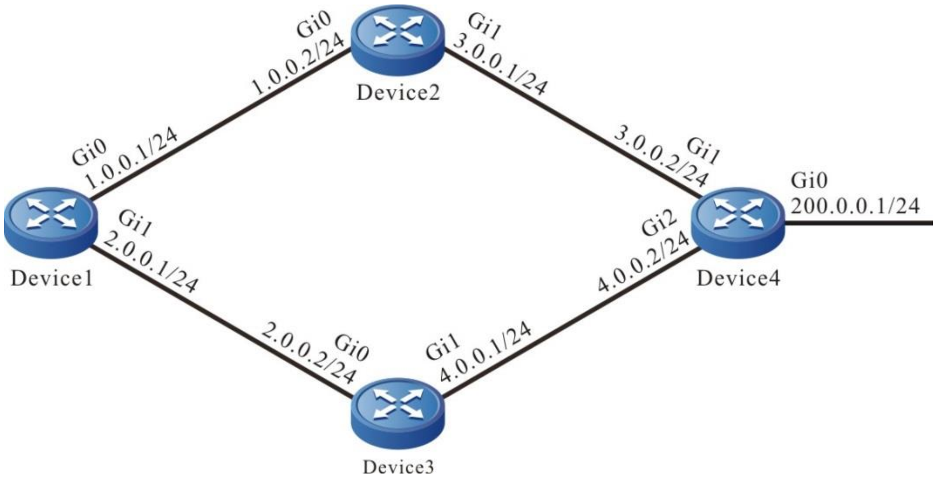

Figure 10–6 Networking of the IRMP unequal-cost load balancing

Configuration Steps

Step 1: Configure the IP address of the interfaces. (Omitted)

Step 2: Configure the IRMP.

#Configure Device1.

```
Device1#configure terminal
Device1(config)#router irmp 100
Device1(config-irmp)#network 1.0.0.0 0.0.0.255
Device1(config-irmp)#network 2.0.0.0 0.0.0.255
Device1(config-irmp)#exit
```

#Configure Device2.

```
Device2#configure terminal
Device2(config)#router irmp 100
Device2(config-irmp)#network 1.0.0.0 0.0.0.255
Device2(config-irmp)#network 3.0.0.0 0.0.0.255
Device2(config-irmp)#exit
```

#Configure Device3.

```
Device3#configure terminal
Device3(config)#router irmp 100
Device3(config-irmp)#network 2.0.0.0 0.0.0.255
Device3(config-irmp)#network 4.0.0.0 0.0.0.255
Device3(config-irmp)#exit
```

#Configure Device4.

```
Device4#configure terminal
Device4(config)#router irmp 100
```

```
Device4(config-irmp)#network 3.0.0.0 0.0.0.255
Device4(config-irmp)#network 4.0.0.0 0.0.0.255
Device4(config-irmp)#network 200.0.0.0 0.0.0.255
Device4(config-irmp)#exit
```

#View the IRMP neighbor information of Device1.

```
Device1#show ip irmp neighbor
IP-IRMP neighbors for process 100 Total neighbor 2
Address        Interface                Hold(s)  Uptime   SeqNum   Srtt(ms)  Rto(s)
1.0.0.2       gigabitethernet0        11      00:10:37  10      0       2
2.0.0.2        gigabitethernet1        12      00:10:15  9       0       2
```

#View the IRMP neighbor information of Device4.

```
Device4#show ip irmp neighbor
IP-IRMP neighbors for process 100 Total neighbor 2
Address        Interface                Hold(s)  Uptime   SeqNum   Srtt(ms)  Rto(s)
3.0.0.1       gigabitethernet1        14      00:11:37  13      0       2
4.0.0.1
gigabitethernet2        12      00:10:45  12      0       2
```

Device1 sets up the IRMP neighbor with Device2, Device3 respectively. Device4 sets up the IRMP neighbor with Device2, Device3 respectively.

#View the topology table and route table of Device1.

```
Device1#show ip irmp topology
IP-IRMP Topology Table for process 100
Codes:  P - Passive, A - Active, H - Holddown, D - Hidden
     > - FIB route, * - FIB successor

P >1.0.0.0/24, 1 successors, FD is 2816
     *via Connected (2816/0), gigabitethernet0P >2.0.0.0/24, 1 successors, FD is 2816
     *via Connected (2816/0), gigabitethernet1
P >3.0.0.0/24, 1 successors, FD is 3072
     *via 1.0.0.2 (3072/2816), gigabitethernet0
P >4.0.0.0/24, 1 successors, FD is 3072
     *via 2.0.0.2 (3072/512), gigabitethernet1
P >200.0.0.0/24, 2 successors, FD is 3328
     *via 2.0.0.2 (3328/3072), gigabitethernet1
     *via 1.0.0.2 (3328/3072), gigabitethernet0
```

```
Device1#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

C   1.0.0.0/24 is directly connected, 13:16:35, gigabitethernet0
L   1.0.0.1/32 is directly connected, 13:16:35, gigabitethernet0
C   2.0.0.0/24 is directly connected, 13:19:24, gigabitethernet1
L   2.0.0.1/32 is directly connected, 13:19:24, gigabitethernet1
E   3.0.0.0/24 [90/3072] via 1.0.0.2, 00:03:22, gigabitethernet0
E   4.0.0.0/24 [90/3072] via 2.0.0.2, 00:22:01, gigabitethernet1
C   127.0.0.0/8 is directly connected, 22:27:25, lo0
L   127.0.0.1/32 is directly connected, 22:27:25, lo0
E   200.0.0.0/24 [90/3328] via 2.0.0.2, 00:21:22, gigabitethernet1
                 [90/3328] via 1.0.0.2, 00:03:22, gigabitethernet0
```

There are two load balancing routing to the network segment 200.0.0.0/24 in Device1 route table. The forwarding paths to the network segment are Device1→Device2→Device4 and Device1→Device3→Device4.

Step 3: Change the interface bandwidth.

#Change the bandwidth of the interface connected to Device1 and Device3 to 100000 kbps.

```
Device1(config)#interface gigabitethernet1
Device1(config-if-gigabitethernet1)#bandwidth 100000
Device1(config-if-gigabitethernet1)#exit
```

#View the topology table and route table of Device1.

```
Device1#show ip irmp topology
IP-IRMP Topology Table for process 100
Codes:  P - Passive, A - Active, H - Holddown, D - Hidden
        > - FIB route, * - FIB successor

P >1.0.0.0/24, 1 successors, FD is 2816
        *via Connected (2816/0), gigabitethernet2/2/0
P >2.0.0.0/24, 1 successors, FD is 25856
        *via Connected (25856/0), gigabitethernet2/2/1
P >3.0.0.0/24, 1 successors, FD is 3072
        *via 1.0.0.2 (3072/2816), gigabitethernet2/2/0
P >4.0.0.0/24, 1 successors, FD is 26112
        *via 2.0.0.2 (26112/2816), gigabitethernet2/2/1
```

```
P >200.0.0.0/24, 2 successors, FD is 3328
       *via 1.0.0.2 (3328/3072), gigabitethernet2/2/0
        via 2.0.0.2 (26368/3072), gigabitethernet2/2/1

Device1#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
      U - Per-user Static route
      O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

C   1.0.0.0/24 is directly connected, 28:44:11, gigabitethernet0
L   1.0.0.1/32 is directly connected, 28:44:11, gigabitethernet0
C   2.0.0.0/24 is directly connected, 00:14:05, gigabitethernet1
L   2.0.0.1/32 is directly connected, 00:14:05, gigabitethernet1
E   3.0.0.0/24 [90/3072] via 1.0.0.2, 00:10:11, gigabitethernet0
E   4.0.0.0/24 [90/28416] via 2.0.0.2, 00:07:57, gigabitethernet1
C   127.0.0.0/8 is directly connected, 220:26:51, lo0
L   127.0.0.1/32 is directly connected, 220:26:51, lo0
E   200.0.0.0/24 [90/3328] via 1.0.0.2, 00:10:11, gigabitethernet0
```

After modifying the interface bandwidth, Device1 preferentially chooses the routing 200.0.0.0/24 advertised by Device2.

Step 4: Configure the IRMP unequal-cost load balancing.

#Configure Device1 and configure the load balancing conversion factor as 100.

```
Device1(config)#router irmp 100
Device1(config-irmp)#variance 100
Device1(config-irmp)#exit
```

For details about the load balancing conversion factor, refer to the load balancing chapter in the IRMP function configuration.

Step 5: Check the result.

#View the topology table and route table of Device1.

```
Device1#show ip irmp topology
IP-IRMP Topology Table for process 100
Codes:  P - Passive, A - Active, H - Holddown, D - Hidden
       > - FIB route, * - FIB successor

P >1.0.0.0/24, 1 successors, FD is 2816
       *via Connected (2816/0), gigabitethernet0
P >2.0.0.0/24, 1 successors, FD is 28160
```

```
     *via Connected (28160/0), gigabitethernet1
P >3.0.0.0/24, 1 successors, FD is 3072
     *via 1.0.0.2 (3072/2816), gigabitethernet0
P >4.0.0.0/24, 1 successors, FD is 28416
     *via 2.0.0.2 (28416/2816), gigabitethernet1
P >200.0.0.0/24, 2 successors, FD is 3328
     *via 1.0.0.2 (3328/3072), gigabitethernet0
     *via 2.0.0.2 (26368/3072), gigabitethernet1

Device1#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
    U - Per-user Static route
    O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

C   1.0.0.0/24 is directly connected, 28:47:15, gigabitethernet0
L   1.0.0.1/32 is directly connected, 28:47:15, gigabitethernet0
C   2.0.0.0/24 is directly connected, 00:17:08, gigabitethernet1
L   2.0.0.1/32 is directly connected, 00:17:08, gigabitethernet1
E   3.0.0.0/24 [90/3072] via 1.0.0.2, 00:13:15, gigabitethernet0
E   4.0.0.0/24 [90/28416] via 2.0.0.2, 00:11:00, gigabitethernet1
C   127.0.0.0/8 is directly connected, 220:29:54, lo0
L   127.0.0.1/32 is directly connected, 220:29:54, lo0
E   200.0.0.0/24 [90/3328] via 1.0.0.2, 00:13:15, gigabitethernet0
           [90/26368] via 2.0.0.2, 00:00:12, gigabitethernet1
```

Routing 200.0.0.0/24 on Device1 forms the unequal-cost load balancing. Data will be transmitted for load balancing on these two paths based on the inverse ratio of the metric value.

11. BGP

11.1. Overview

Border Gateway Protocol (BGP) is a routing protocol that exchanges Network Layer Reachability Information (NLRI) between Autonomous Systems (ASs). Internal Gateway Protocols (IGPs) such as Routing Information Protocol (RIP), Open Shortest Path First (OSPF), and Intermediate System-to-Intermediate System (IS-IS), focus mainly on finding accurate paths, taking network nodes (such as routers, layer-3 switches, multi-NIC hosts) as the routing units. Different from IGPs, External Gateway Protocols (EGPs) focuses mainly on controlling the routing direction, taking AS networks as the routing units.

BGP is used for interconnection between AS networks. It supports routing information exchange between ASs. It is usually used for large-scale network aggregation and network core. Its application layer determines that BGI has the following features when compared with IGP:

- BGP uses the TCP protocol to transmit packets through service port 179. TCP ensures the reliability of transmission, so BGP need not provide an independent transmission control policy for reliable transmission of information.
- BGP updates routes in incremental mode, that is, it informs its neighbors of route changes only when route properties are changed, or a route is added or deleted. This mode greatly decreases network bandwidth that is occupied by BGP in transmitting routes.
- BGP is an AS-based distance vector protocol. It carries AS path properties in routing packets to solve the routing loop problem.
- BGP routes have abundant properties. You can modify the properties by applying a routing policy. In this way, you can control route filtering and selection freely.
- BGP has two neighbor types, Interior Border Gateway Protocol (IBGP) and External Border Gateway Protocol (EBGP). Between different types of neighbors, different route advertisements and routing policies are used.

11.2. BGP Function Configuration

Table 11-1 BGP function list

Configuration Tasks	
Configure a BGP neighbor.	Configure an IBGP neighbor.
	Configure an EBGP neighbor.
	Configure a BGP passive neighbor.
	Configure an MP-BGP neighbor.
	Configure MD5, SM3 or KEYCHAIN authentication for BGP neighbors.

Configuration Tasks	
Configure BGP route generation.	Configure BGP to advertise local routes.
	Configuring BGP to redistribute routes.
	Configure BGP to advertise the default route.
Configure BGP route control.	Configure BGP to advertise aggregated routes.
	Configure the administrative distance of BGP routes.
	Configure routing policies in the outgoing direction of a BGP neighbor.
	Configure a routing policies in the incoming direction of a BGP neighbor.
	Configure the maximum number of routes that a BGP neighbor receives.
	Configure the maximum number of BGP load balancing routes.
Configure BGP route properties.	Configure the BGP route weight.
	Configure the MED property of a BGP route.
	Configure the Local-Preference property of a BGP route.
	Configure the AS_PATH property of a BGP route.
	Configure the NEXT-HOP property of a BGP route.
	Configure the community property of a BGP route.

Configuration Tasks	
Configure BGP network optimization.	Configure the keep-alive time of BGP neighbors.
	Configure BGP route detection time.
	Configure quick disconnection of EBGP neighbors.
	Configure the BGP route suppression function.
	Configure the BGP neighbor refresh capability.
	Configure the BGP neighbor soft reset capability.
	Configure the ORF capability of BGP neighbors.
Configure a large-scale BGP network.	Configure a BGP peer group.
	Configure a BGP route reflector.
	Configure a BGP confederation.
Configure BGP to coordinate with BFD.	Configure EBGP to coordinate with BFD.
	Configure IBGP to coordinate with BFD.
Configure BGP fast re-routing	Configure BGP fast re-routing
Configure BGP LS	Configure BGP LS

11.2.1. Configure BGP Neighbor

Configuration Condition

Before configuring a BGP neighbor, ensure that:

- The link layer protocol has been configured to ensure normal communication at the link layer.
- The network layer addresses of the interfaces have been configured so that the adjacent network nodes are reachable at the network layer.

Configure an IBGP Neighbor

1. Perform basic configuration.

In configuring an IBGP neighbor, you need to set the AS of the neighbor to be the same as the AS of the local device. You can configure a Router ID for a device. The Router ID is used to uniquely identify a BGP device in setting up a BGP session. If no Router ID is configured for a device, the device selects a Router ID from interface addresses. The rules for selection are as follows:

- Select the biggest IP address from loopback interface IP addresses as the Router ID.
- If no loopback interface is configured with an IP address, select the biggest IP address from the IP addresses of other interfaces as the Router ID.
- Only when an interface is in the UP status can the IP address of the interface be elected as the Router ID.

Table 11-2 Configure an IBGP neighbor

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enable the BGP protocol and enter the BGP configuration mode.	**router bgp** *autonomous-system*	Mandatory. By default, BGP is disabled.
Configure a Router ID for the BGP device.	**bgp router-id** *router-id*	Optional. By default, the device selects a Router ID from interface addresses. The loopback interface and large IP address have the priorities.
Configure an IBGP neighbor.	**neighbor** { *neighbor-address* | *peer-group-name* } **remote-as** *as-number*	Mandatory. By default, no IBGP neighbor is created.
Activate the capability of an IBGP neighbor in transmitting and receiving IPv4 unicast routes.	**neighbor** { *neighbor-address* | *peer-group-name* } **activate**	Optional. By default, the IBGP neighbor's capability in transmitting and receiving IPv4 unicast routes is activated automatically.

Step	Command	Description
Configure a description for an IBGP neighbor.	**neighbor** { *neighbor-address* | *peer-group-name* } **description** *description-string*	Optional. By default, no description is configured for an IBGP neighbor.

2. Configure the source address of a TCP session.

BGP uses the TCP protocol to transmit packets through service port 179. TCP features reliable transmission, ensuring that BGP protocol packets can be properly transmitted to its neighbors.

Table 11-3 Configure the source address of a TCP session

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enable the BGP protocol and enter the BGP configuration mode.	**router bgp** *autonomous-system*	-
Configure an IBGP neighbor.	**neighbor** { *neighbor-address* | *peer-group-name* } **remote-as** *as-number*	Mandatory. By default, no IBGP neighbor is created.
Configure the source address of a TCP session of an IBGP neighbor.	**neighbor** { *neighbor-address* | *peer-group-name* } **update-source** { *interface-name* | *ip-address* }	Mandatory. By default, the TCP session automatically selects the address of a routing output interface as the source address.

Note:

- If there exist load balancing routes, the source addresses must be configured for TCP sessions of BGP neighbors. If TCP session source addresses are not configured, if the neighbors have different optimal routes, they may use different output interfaces as their source addresses. In this way, BGP sessions may fail to set up within a period of time.

Configure an EBGP Neighbor

1. Perform basic configuration.

In configuring an EBGP neighbor, you need to set the AS of the neighbor to be different from the AS of the local device.

Table 11-4 Configure an EBGP neighbor

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enable the BGP protocol and enter the BGP configuration mode.	**router bgp** *autonomous-system*	-
Configure an EBGP neighbor.	**neighbor** { *neighbor-address* | *peer-group-name* } **remote-as** *as-number*	Mandatory. By default, no EBGP neighbor is created.

2. Configure a non-direct-connect EBGP neighbor

EBGP neighbors are located in different operation networks, and they are usually connected by a direct-connect physical link. Therefore, the default TTL value for the IP packets between EBGP neighbors is 1. In non-direct-connect operation networks, you can use a command to set the TTL value of IP packets so as to set up a BGP connection.

Table 11-5 Configure a non-direct-connect EBGP neighbor

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-
Configure an EBGP neighbor.	**neighbor** { *neighbor-address* | *peer-group-name* } **remote-as** *as-number*	Mandatory. By default, no EBGP neighbor is created.
Configure the source address of a TCP session of an EBGP neighbor.	**neighbor** { *neighbor-address* | *peer-group-name* } **update-source** { *interface-name* | *ip-address* }	Optional. By default, the TCP session automatically selects the address of a routing output interface as the source address.

Step	**Command**	**Description**
Allow non-direct-connect EBGP neighbors to set up a connection.	**neighbor** { *neighbor-address* | *peer-group-name* } **ebgp-multihop** [*ttl-value*]	Mandatory. By default, non-direct-connect devices are not allowed to form EBGP neighbors.

Configure a BGP Passive Neighbor

In some special application environments, the BGP passive neighbor function is in need. After the passive neighbor function is enabled, the BGP does not initiate the TCP connection request for setting up a BGP neighbor relation; instead, it waits for the neighbor's connection request before setting up a neighbor relation. By default, neighbors initiate connection requests to each other. If connections conflict, they select an optimal TCP connection to form a BGP session. Before configuring a BGP passive neighbor, you need to configure a BGP neighbor.

Table 11-6 Configure a BGP passive neighbor

Step	**Command**	**Description**
Enter the global configuration mode.	**configure terminal**	-
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-
Configure a BGP neighbor.	**neighbor** { *neighbor-address* | *peer-group-name* } **remote-as** *as-number*	Mandatory. By default, no BGP neighbor is created.
Configure a BGP passive neighbor.	**neighbor** { *neighbor-address* | *peer-group-name* } **passive**	Mandatory. By default, no passive neighbor is activated.

Configure an MP-BGP Neighbor

By default, BGP neighbors are activated in the IPv4 unicast address family, and they have the capability of transmitting and receiving IPv4 unicast routes. Neighbors need to be enabled by using a command in other address families, such as multicast address family, VRF address family, VPN address family, and MVPN address family so that they have the capability of transmitting and receiving the required routes. Before configuring an MP-BGP neighbor, you need to configure a BGP neighbor.

Table 11-7 Configure an MP-BGP neighbor

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-
Configure a BGP neighbor.	**neighbor** { *neighbor-address* | *peer-group-name* } **remote-as** *as-number*	Mandatory. By default, no BGP neighbor is created.
Enter the BGP IPv4 configuration mode.	**address-family ipv4 multicast**	Mandatory. By default, after entering the BGP configuration mode, the user is in unicast address family mode.
Activate neighbors in BGP IPv4 multicast address family.	**neighbor** { *neighbor-address* | *peer-group-name* } **activate**	Mandatory. By default, global neighbors are deactivated in multicast address family mode.
Exit the BGP IPv4 configuration mode.	**exit-address-family**	-
Enter the BGP IPv4 VRF configuration mode.	**address-family ipv4 vrf** *vrf-name*	-
Configure neighbors in BGP IPv4 VRF address family mode.	**neighbor** { *neighbor-address* | *peer-group-name* } **remote-as** *as-number*	Mandatory. By default, no BGP neighbor is created.

Step	Command	Description
Activate neighbors in IPv4 VRF address family mode.	**neighbor** { *neighbor-address* | *peer-group-name* } **activate**	Optional. By default, neighbors are activated in BGP IPv4 VRF configuration mode.
Exit the BGP IPv4 VRF configuration mode.	**exit-address-family**	-
Enter the BGP VPNv4 config mode.	**address-family vpnv4** [**unicast**]	-
Activate neighbors in BGP VPNv4 address family mode.	**neighbor** { *neighbor-address* | *peer-group-name* } **activate**	Mandatory. By default, global neighbors are deactivated in VPN address family mode.
Exit the BGP VPNv4 configuration mode.	**exit-address-family**	-
Enters the BGP MDT configuration mode.	**address-family ipv4 mdt**	-
Activate neighbors in BGP MDT address family mode.	**neighbor** { *neighbor-address* | *peer-group-name* } **activate**	Mandatory. By default, global neighbors are deactivated in MDT address family mode.
Exit the BGP MDT configuration mode	**exit-address-family**	-
Enter the BGP MVPN configuration mode	**address-family ipv4 mvpn**	-

Step	Command	Description
Activate neighbors in the BGP MVPN address family	**neighbor** { *neighbor-address* | *peer-group-name* } **activate**	Mandatory By default, the global neighbors are not activated in the MVPN address family.
Exit the BGP MVPN configuration mode	**exit-address-family**	-

Note:

- The neighbors that are configured in BGP configuration mode and BGP IPv4 unicast configuration mode are global neighbors, and the neighbors that are configured in BGP IPv4 VRF configuration mode belong only to the VRF address family.

Configure MD5, SM3 or KEYCHAIN Authentication for BGP Neighbors

BGP supports configuring MD5, SM3 or KEYCHAIN authentication to protect information exchange between neighbors. MD5, SM3 or KEYCHAIN authentication is implemented by the TCP protocol. Two neighbors must be configured with the same authentication password before a TCP connection can be set up; otherwise, if the TCP protocol fails in MD5, SM3 or KEYCHAIN authentication, the TCP connection cannot be set up. Before configuring MD5, SM3 or KEYCHAIN authentication for BGP neighbors, you need to configure BGP neighbors.

Table 11-8 Configure MD5, SM3 or KEYCHAIN authentication for BGP neighbors

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-
Configure a BGP neighbor.	**neighbor** { *neighbor-address* | *peer-group-name* } **remote-as** *as-number*	Mandatory. By default, no BGP neighbor is created.
Configure the MD5, SM3 or KEYCHAIN authentication for BGP neighbors.	**neighbor** { *neighbor-address* | *peer-group-name* } { **password** [**sm3**] [**0** | **7**]*password-string* | keychain *keychain-name* }	Mandatory. By default, no MD5, SM3 or KEYCHAIN authentication is started for BGP neighbors.

11.2.2. Configure BGP Route Generation

Configuration Condition

Before configuring BGP route generation, ensure that:

- BGP is enabled.
- BGP neighbors are configured and a session is set up successfully.

Configure BGP to Advertise Local Routes

BGP can use the **network** command to introduce the routes of the IP route table into the BGP route table. Only when there are routes that match completely the **network** prefix and mask can the routes be introduced into the BGP route table and advertised.

In advertising a local route, you can apply a route map for the route, and you can also specify the route as the backdoor route. The backdoor route takes EBGP routes as local BGP routes and uses the administrative distance of local routes. This allows IGP routes to have higher priorities than EBGP routes. At the same time, backdoor routes will not be advertised to EBGP neighbors.

Table 11-9 Configure BGP to advertise local routes

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-
Configure BGP to advertise local routes.	**network** *ip-address mask* [**route-map** *rtmap-name* [**backdoor**] | **backdoor**]	Mandatory. By default, BGP does not advertise local routes.

Note:

- The Origin property type of the local routes that are advertised by BGP is IGP.
- If you run the **network backdoor** command for an EBGP route, the administrative distance of the EBGP route changes to the local route administrative distance. (By default, the EBGP route administrative distance is 20, and the local route administrative distance is 200.) Then, the administrative distance of the EBGP route is smaller than the administrative distance of the default IGP route. In this way, the IGP route is selected with priority, forming a backdoor link between EBGP neighbors.
- The route map applied to the local routes that are advertised by BGP supports match options, including as-path, community, extcommunity, ip address, ip nexthop, and metric, and it supports the set options, including as-path, comm-list, community, extcommunity, ip next-hop, local-preference, metric, origin, and weight.

Configure BGP to Redistribute Routes

BGP is not responsible for route learning. It focuses mainly on managing route properties so as to control the route direction. Therefore, BGP redistributes IGP routes to generate BGP routes and advertise the BGP routes to neighbors. When BGP redistributes IGP routes, it can apply a routing diagram.

Table 11-10 Configure BGP to redistribute routes

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-
Configure BGP to redistribute IGP routes.	**redistribute** { **connected** | **irmp** *as-number* | **isis** [*area-tag*] [**match** *isis-level*] | **ospf** *as-number* [**match** *route-sub-type*] | **rip** | **static** } [**route-map** *map-name* / **metric** *value*]	Mandatory. By default, BGP does not redistribute IGP routes.

Note:

- The Origin property type of the IGP routes that are advertised by BGP is INCOMPLETE.
- The route map applied to other protocol routes that are redistributed by BGP supports match options, including as-path, community, extcommunity, ip address, ip nexthop, and metric, and it supports the set options, including as-path, comm-list, community, extcommunity, ip next-hop, local-preference, metric, origin, and weight.

Configure BGP to Advertise the Default Route

Before BGP advertises a default route to neighbors, the default route needs to be introduced. Two ways of introducing the default routes are available: Running the **neighbor default-originate** command to generate a BGP default route, and running the **default-information originate** command to redistribute the default route of another protocol.

The default route that is generated by running the **neighbor default-originate** command is route 0.0.0.0/0 that is automatically generated by BGP. The default route that is redistributed by running the **default-information originate** command is route 0.0.0.0/0 of the redistributed protocol introduced by BGP.

Table 11-11 Configure BGP to advertise the default route

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-

Step	Command	Description
Configure BGP to generate the default route.	**neighbor** { *neighbor-address* | *peer-group-name* } **default-originate** [**route-map** *rtmap-name*]	Mandatory. By default, BGP does not generate the default route.
Configure the default route of another protocol redistributed by BGP.	**default-information originate**	Mandatory. By default, BGP does not redistribute the default route of another protocol.

Note:

- In configuring BGP to redistribute the default route of another protocol, you need to configure BGP to redistribute routes.
- In configuring BGP to generate a default route, you can apply a route map to the route.
- The route map that is applied to the default route that is generated by BGP supports set options, including as-path, comm-list, community, extcommunity, ip next-hop, local-preference, metric, origin, and weigh.

11.2.3. Configure BGP Route Control

Configuration Condition

Before configuring BGP route control, ensure that:

- BGP is enabled.
- BGP neighbors are configured and a session is set up successfully.

Configure BGP to Advertise Aggregated Routes

In a large-scale BGP network, to decrease the number of routes that are advertised to neighbors or effectively control BGP routing, you can configure a BGP aggregated route.

Table 11-12 Configure BGP to advertise aggregated routes

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-

Step	Command	Description
Configure BGP to advertise aggregated routes.	**aggregate-address** *ip-address mask* [**as-set** / **summary-only** / **route-map** *rtmap-name*]	Mandatory. By default, BGP does not aggregate routes.

Note:

- When configuring BGP to advertise aggregated routes, you can specify the **summary-only** option so that BGP advertises only aggregated routes. This decreases the number of routes that are advertised.
- You can specify the **as-set** option to generate aggregation routes with the AS_PATH property.
- You can also apply a route map to the aggregation routes so as to set more abundant properties for the aggregation routes.

Configure the Administrative Distance of BGP Routes

In the IP route table, each protocol controls the administrative distance of routing. The smaller the administrative distance is, the higher the priority is .BGP affects routing by specifying the administrative distances of specified network segments. The administrative distances of the routes that cover the specified network segments will be modified. Meanwhile, ACL is applied to filter the network segments that are covered by the routes, that is, only the administrative distances of the network segment that are allowed by the ACL can be modified.

The **distance bgp** command is used to modify the management distances of external, internal, and local BGP routes. The **distance** command is only used to modify the administrative distances of specified network segments. The **distance** command has a higher priority than the **distance bgp** command. The network segments that are covered by the **distance** command use the administrative distance that is specified by the command, while the network segments that are not covered by the distance command use the administrative distance that is specified by the **distance bgp** command.

Table 11-13 Configure the administrative distance of a BGP route

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-
Configure BGP to modify the default administrative distance.	**distance bgp** *external-distance internal-distance local-distance*	Optional. By default, the administrative distance

Step	Command	Description
Configure the administrative distance of a specified network segment.	**distance** *administrative-distance ip-address mask* [*acl-name*]	of EBGP routes is 20, the administrative distance of IBGP routes is 200, and the administrative distance of local routes is 200.

Configure Routing Policies in the Outgoing Direction of a BGP Neighbor

BGP route advertisement or routing is implemented based on the powerful routing properties. When advertising routes to neighbors, you can apply routing policies to modify route properties or filter some routes. Currently, the routing policies that can be applied in the outgoing direction include:

- distribute-list: Distribution list.
- filter-list: AS_PATH property filtration list.
- prefix-list: IP prefix list.
- route-map: Route map.

Table 11-14 Configure routing policies in the outgoing direction of a BGP neighbor

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-
Apply the distribution list in the outgoing direction.	**neighbor** { *neighbor-address* | *peer-group-name* } **distribute-list** *access-list-name* **out**	You can select multiple options. (However, the distribution list and the IP prefix list cannot be configured at the same time.) By default, no routing policy is configured in the outgoing direction of a BGP neighbor.
Apply the AS_PATH property filtration list in the outgoing direction.	**neighbor** { *neighbor-address* | *peer-group-name* } **filter-list** *aspath-list-name* **out**	
Apply the IP prefix list in the outgoing direction.	**neighbor** { *neighbor-address* | *peer-group-name* } **prefix-list** *prefix-list-name* **out**	

Step	Command	Description
Apply a route map in the outgoing direction.	**neighbor** { *neighbor-address* | *peer-group-name* } **route-map** *rtmap-name* **out**	

Note:

- After configuring the routing policy in the outgoing direction of a BGP neighbor, you need to reset the neighbor to validate the settings.
- If you apply a route map in the outgoing direction of a route reflector, this changes only the NEXT-HOP property.
- For how to configure a filtration list, refer to the "Configure AS-PATH" section of the "Routing Policy Tools" chapter.
- In configuring routing policies in the outgoing direction of a BGP neighbor, you can configure multiple policies at the same time. BGP applies routing policies in the sequence of **distribute-list**, **filter-list**, **prefix-list**, and **route-map**. If a former policy is rejected, the latter policies will not be applied. The routing information can be advertised only after it passes all the configured policies.
- The route map that is applied in the outgoing direction of a BGP route supports match options, including as-path, community, extcommunity, ip address, ip nexthop, and metric, and it supports the set options, including as-path, comm-list, community, extcommunity, ip next-hop, local-preference, metric, origin, and weight.

Configure Routing Policies in the Incoming Direction of a BGP Neighbor

BGP can apply routing policies to filter received routing information or modify route properties. Similar to the policies applied in the outgoing directions, four polices are applied in the incoming directions:

- distribute-list: Distribution list.
- filter-list: AS_PATH property filtration list.
- prefix-list: IP prefix list.
- route-map: Route map.

Table 11-15 Configure Routing Policies in the Incoming Direction of a BGP neighbor

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-

Step	Command	Description
Apply the distribution list in the incoming direction.	**neighbor** { *neighbor-address* | *peer-group-name* } **distribute-list** *access-list-name* **in**	You can select multiple options. (However, the distribution list and the IP prefix list cannot be configured at the same time.) By default, no policy is applied in the incoming direction.
Apply the AS_PATH property filtration list in the incoming direction.	**neighbor** { *neighbor-address* | *peer-group-name* } **filter-list** *aspath-list-name* **in**	
Apply the IP prefix list in the incoming direction.	**neighbor** { *neighbor-address* | *peer-group-name* } **prefix-list** *prefix-list-name* **in**	
Apply a route map in the incoming direction.	**neighbor** { *neighbor-address* | *peer-group-name* } **route-map** *rtmap-name* **in**	

Note:

- After configuring the routing policy in the incoming direction of a BGP neighbor, you need to reset the neighbor to validate the settings.
- In configuring routing policies in the incoming direction of a BGP neighbor, you can configure multiple policies at the same time. BGP applies routing policies in the sequence of **distribute-list**, **filter-list**, **prefix-list**, and **route-map**. If a former policy is rejected, the latter policies will not be applied. A route can be added into the database after it passes all the configured policies.
- The routing policies applied in the incoming direction of a BGP route support match options, including as-path, community, extcommunity, ip address, ip nexthop, and metric, and they support the set options, including as-path, comm-list, community, extcommunity, ip next-hop, local-preference, metric, origin, and weight.

Configure the Maximum Number of Routes that a Device Receives

The BGP device supports limiting the number of the received routes. When the number of the routes reaches the threshold, print the log alarm, and do not receive the route any more. When the number of the routes is smaller than the threshold, re-receive the route again.

Table 11-16 Configure the maximum number of routes that a device Receives

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-

Step	Command	Description
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-
Configure the maximum number of the routes that the device can receive	**maximum-prefix** *{prefix-num}*	Mandatory By default, the threshold is 30000.

Configure the Maximum Number of Routes that a BGP Receives from a Neighbor

You can limit the number of routes that a BGP receives from a specified neighbor. Once the number of routes the BGP receives from the neighbor reaches a threshold, an alarm is generated or the neighbor is disconnected.

Table 11-17 Configure the maximum number of routes that a BGP Receives from a neighbor

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-
Configure the maximum number of routes that a BGP receives from a neighbor.	**neighbor** { *neighbor-address* | *peer-group-name* } **maximum-prefix** *prefix-num* [*threshold-value*] [**warning-only**]	Mandatory. By default, the number of routes that a BGP receives from a neighbor is not limited.

Note:

- If the warning-only option is not specified, after the number of routes that the BGP receives from the neighbor reaches the maximum number, the BGP session is automatically disconnected.
- If the warning-only option is specified, after the number of routes that the BGP receives from the neighbor reaches the maximum number, a warning message is displayed, but route learning continues.

Configure the Maximum Number of BGP Load Balancing Routes

In a BGP networking environment, if several paths with the same cost are available to reach the same destination, you can configure the number of BGP load balancing routes for load balancing.

Table 11-18 Configure the Maximum number of BGP load balancing routes

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-
Configure the maximum number of IBGP load balancing routes.	**maximum-paths ibgp** *number*	Mandatory. By default, IBGP does not support load balancing routes.
Configure the maximum number of EBGP load balancing routes.	**maximum-paths** *number*	Mandatory. By default, EBGP does not support load balancing routes.
Configure the maximum number of the load balance routes of IBGP and EBGP	**maximum-paths eibgp** *number*	Mandatory By default, EBGP and IBGP routes cannot perform the load balance routing.

Note:

- After the maximum number of EBGP load balancing routes is configured, load balancing takes effect only when EBGP routes are selected with priority.
- In different BGP configuration modes, the commands for configuring the maximum number of load balancing routes are different. For details, refer to the description of **maximum-paths** in the BGP command manual.

11.2.4. Configure BGP Route Properties

Configuration Condition

Before configuring BGP route properties, ensure that:

- BGP is enabled.
- BGP neighbors are configured and a session is set up successfully.

Configure the BGP Route Weight

In BGP routing, the first rule is to compare the weights of routes. The larger the weight of a route is, the higher the priority it has. The weight of a route is the local property of the device, and it cannot be transferred to other BGP neighbors. The value range of a route weight is 1-65535. By default, the weight of a route that has been learnt from a neighbor is 0, and the weights of all routes that are generated by the local device are all 32768.

Table 11-19 Configure the BGP route weight

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-
Configure the weight of a route of a neighbor or peer group.	**neighbor** { *neighbor-address* | *peer-group-name* } **weight** *weight-num*	Mandatory. By default, the weight of a route of a neighbor is 0.

Configure the MED Property of a BGP Route

Multi-Exit Discriminator (MED) properties are used to select the optimal route for the traffic that enters an AS. If the other routing conditions are the same and BGP learns several routes with the same destination from different EBGP neighbors, BGP select the route with the minimum MED value as the optimal ingress.

MED sometimes is also called external metric. It is marked as a "metric" in the BGP route table. BGP advertises the MED properties of the routes that it has learnt from neighbors to IBGP neighbors, but BGP does not advertise the MED properties to EBGP neighbors. Therefore, MED properties are applicable to only adjacent ASs.

1. Configure BGP to allow comparing MEDs of neighbor routes from different ASs.

By default, BGP implements MED route selection only among the routes that are from the same AS. However, you can run the **bgp always-compare-med** command to let BGP ignore the limitation on the same AS in MED route selection.

Table 11-20 Configure BGP to allow comparing MEDs of neighbor routes from different ASs

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-

Step	Command	Description
Configure BGP to allow comparing MEDs of neighbor routes from different ASs.	**bgp always-compare-med**	Mandatory. By default, BGP allows only to compare MEDs of neighbor routes from the same AS.

2. Configure BGP to sort and select MEDs according to AS_PATH groups.

By default, BGP is not enabled to sort and select MEDs according to route AS_PATH groups. To enable the function, run the **bgp deterministic-med** command. In route selection, all routes are organized based on AS_PATHs. In each AS_PATH group, routes are sorted based MED values. The route with the minimum MED value is selected as the optimal route in the group.

Table 11-21 Configure BGP to sort and select MEDs according to AS_PATH groups

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-
Compare confederation MEDs in BGP route selection.	**bgp deterministic-med**	Mandatory. By default, confederation MEDs are not compared in BGP route selection.

3. Configure to compare MEDs of routes in the local confederation.

By default, the MED values of EBGP routes from different ASs are not compared. The setting is valid for the EBGP routes of confederations. To enable comparison of MED values of routes of the local confederation, run the **bgp bestpath med confed** command.

Table 11-22 Configure BGP to compare MEDs of routes in the local confederation

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-

Step	Command	Description
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-
Configure BGP to compare MED values of routes in the local confederation.	**bgp bestpath med confed**	Mandatory. By default, the MED values of routes in the local confederations will not be compared.

4. Configure a route map to modify MED properties.

In transmitting and receiving routes, you can apply a route map to modify MED properties.

Table 11-23 Configure a route map to modify MED properties

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-
Configuring a route map to modify MED properties.	**neighbor** { *neighbor-address* | *peer-group-name* } **route-map** *rtmap-name* **in | out**	Mandatory. By default, no route map is applied to any neighbor.

Note:

- In configuring a route map to modify an MED property, you can use the **set metric** command to modify the MED property. For details, refer to Routing Policy Tools-Command Manual-set metric.
- After the **neighbor attribute-unchanged** command is configured, the MED properties of neighbors cannot be changed by the route map that is applied.

Configure the Local-Preference Property of a BGP Route

Local-Preference properties are transferred only between IBGP neighbors. Local-Preference is used to select the optimal egress of an AS. The route with the maximum Local-Preference will be selected with priority.

The value range of Local-Preference is 0-4294967295. The larger the value is, the higher priority the route has. By default, the Local-Preference value of all the routes that are advertised to IBGP neighbors is 100. You can use the **bgp default local-preference** command or the route map to modify the Local-Preference property value.

1. Configure BGP to modify the default Local-Preference property.

Table 11-24 Configure BGP to modify the default local-preference property

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-
Configure the default value of BGP Local-Preference property.	**bgp default local-preference** *local-value*	Optional. By default, the Local-Preference value is 100.

2. Configure the route map to modify the Local-Preference property.

Table 11-25 Configure the route map to modify the local-preference property

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-
Configure the route map to modify the Local-Preference property.	**neighbor** { *neighbor-address* | *peer-group-name* } **route-map** *rtmap-name* **in | out**	Mandatory. By default, the route map is not applied to any neighbor.

Note:

- In configuring a route map to modify the Local-Preference property, you can use the **set local-preference** command to modify the Local-Preference property. For details, refer to Routing Policy Tools-Command Manual-set local-preference.

Configure the AS_PATH Property of a BGP Route

1. Configure BGP to ignore AS_PATHs in route selection.

If the other conditions are the same, BGP selects the route with the shortest AS-PATH in route selection. To cancel route selection based on AS_PATHs, run the **bgp bestpath as-path ignore** command.

Table 11-26 Configure BGP to ignore AS_PATHs in route selection

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-
Configure BGP to ignore AS_PATHs in route selection.	**bgp bestpath as-path ignore**	Mandatory. By default, the AS_PATH values are compared in route selection.

2. Configure the number of local ASs that BGP allows to repeat.

To prevent routing loops, BGP checks the AS_PATH properties of the routes that are received from neighbors, and the routes containing the local AS number will be discarded. However, you can run the **neighbor allowas-in** command to allow the AS_PATH properties of the routes that the BGP receives to contain the local AS number, and you can configure the number of ASs that can be contained.

Table 11-27 Configure the number of local ASs that BGP allows to repeat

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-
Configure the number of ASs that are allowed to repeat.	**neighbor** { *neighbor-address* | *peer-group-name* } **allowas-in** [*as-num*]	Mandatory. By default, the AS_PATH properties of the routes that are received from neighbors do not allow the local AS number.

3. Configure BGP to remove the private AS number when advertising routes to neighbors.

In a large-scale BGP network, the AS_PATH properties of routes contain federation or community property. By default, BGP provides the private AS properties when it advertises routes to neighbors. To mask private network information, run the **neighbor remove-private-AS** command to remove the private AS number.

Table 11-28 Configure BGP to Remove the Private AS number when advertising routes to neighbors

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-
Configure BGP to remove the private AS number when advertising routes to neighbors.	**neighbor** { *neighbor-address* | *peer-group-name* } **remove-private-AS**	Mandatory. By default, when BGP advertise routes to neighbors, it provides the private AS number.

4. Configure to check the validity of the first AS number of an EBGP route.

When BGP advertises a route to EBGP neighbors, it compresses the local AS number to the starting position of the AS_PATH, and the AS that advertises the route first is located at the end. Usually, the first AS of a route that EBGP receives must be the same as the neighbor AS number; otherwise, the route will be discarded.

Table 11-29 Configure to check the validity of the first AS number of an EBGP route

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-
Configure to check the validity of the first AS number of an EBGP route.	**bgp enforce-first-as**	Mandatory. By default, BGP does not enable the mechanism for checking the first AS number.

5. Configure a route map to modify AS_PATH properties.

BGP supports configuring a route map to modify AS_PATH properties. You can run the **set as-path prepend** command to add more routing properties or replace the AS_PATH properties via the **set as-path replace** command so as to affect neighbor routing. In using the **set as-path**

prepend function, first use the local AS to add AS_PATH. If you use another AS, the AS must be emphasized to prevent the AS from rejecting routes that are advertised to it.

Table 11-30 Configure a Route map to modify AS_PATH properties

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-
Configure a route map to modify AS_PATH properties.	**neighbor** { *neighbor-address* | *peer-group-name* } **route-map** *rtmap-name* **in | out**	Mandatory. By default, no route map is applied to any neighbor.

Note:

- In configuring a route map to modify an AS_PATH property, you can use the **set as-path prepend** command or the **set as-path replace** command to modify the AS_PATH property. For details, refer to Routing Policy Tools-Command Manual-**set as-path**.

Configure the NEXT-HOP Property of a BGP Route

When BGP advertises routes to IBGP neighbors, it does not change the routing properties (including the NEXT-HOP property). When BGP advertises the routes that are learned from EBGP neighbors to IBGP neighbors, you can run the **neighbor next-hop-self** command to modify the next-hop property of the routes advertised to BGP neighbors to the local IP address. You can apply a route map to modify the next hop property.

1. Configure BGP to use the local IP address as the next hop of a route.

Table 11-31 Configure BGP to use the local IP address as the next hop of a route

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-

Step	Command	Description
Configure BGP to use the local IP address as the next hop when advertising routes.	**neighbor** { *neighbor-address* | *peer-group-name* } **next-hop-self**	Mandatory. By default, the next-hop property of the routes that are advertised to EBGP neighbors is set to the local IP address, and the next-hop property of the routes that are advertised to IBGP neighbors keeps unchanged.

Note:

- When BGP is configured to use the local IP address as the next hop of a route, if you run the **neighbor update-source** command to configure the source address of a TCP session, the source address is used as the next hop address; otherwise, the IP address of the output interface of the advertising device is selected as the local IP address.

2. Configure a route map to modify NEXT-HOP properties.

BGP supports configuring a route map to modify NEXT-HOP properties. You can run the **set ip next-hop** command to modify the next hop property.

Table 11-32 Configure a route map to modify NEXT-HOP properties

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-
Configure a route map to modify NEXT-HOP properties.	**neighbor** { *neighbor-address* | *peer-group-name* } **route-map** *rtmap-name* **in | out**	Mandatory. By default, no route map is applied to any neighbor.

Note:

- In configuring a route map to modify an NEXT-HOP property, you can use the **set ip next-hop** command to modify the NEXT-HOP property. For details, refer to Routing Policy Tools-Command Manual-**set ip next-hop**.

Configure the Community Property of a BGP Route

When BGP advertises routes to neighbors, it can be configured to send the community property. You can apply a route map to a specified neighbor in the incoming and outgoing directions to match the community properties.

Community property is used to identify a group of routes so as to apply a routing policy to the group of routes. Two types of community property are available: standard and extended. The standard community property consist of 4 bytes, providing the properties such as NO_EXPORT, LOCAL_AS, NO_ADVERTISE, and INTERNET. The extended property consist of eight bytes, providing Route Target (RT) and Route Origin (RO) properties.

1. Configure BGP to advertise route community property to neighbors.

The **neighbor send-community** enables you to advertise standard community property or extended community property or both types of property to neighbors.

Table 11-33 Configure BGP to advertise route community property to neighbors

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-
Configure BGP to advertise route community property to neighbors.	**neighbor** { *neighbor-address* | *peer-group-name* } **send-community** [**both** | **extended** | **standard**]	Mandatory. By default, the community property is not advertised to any neighbor.

Note:

- After neighbors are activated in VPNv4/MVPN address family, standard and extended community properties are automatically advertised to neighbors.

2. Configure a route map to modify the community property.

BGP supports configuring a route map to modify the route community property. You can use the **set communtiy** to command to modify the community property.

Table 11-34 Configure a route map to modify the community property

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-

Step	Command	Description
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-
Configure a route map to modify the BGP route community property	**neighbor** { *neighbor-address* | *peer-group-name* } **route-map** *rtmap-name* **in** | **out**	Mandatory. By default, no route map is applied to any neighbor.

Note:

- In configuring a route map to modify community property, you can use the **set community** command to modify the community property. For details, refer to Routing Policy Tools-Command Manual-set community.

11.2.5. Configure BGP Network Optimization

Configuration Condition

Before configuring BGP network optimization, ensure that:

- BGP is enabled.
- BGP neighbors are configured and a session is set up successfully.

Configure the Keep-alive Time of BGP Neighbors

After a BGP session is successfully set up, keep-alive messages are sent periodically between the neighbors to maintain the BGP session. If no keep-alive message or Update packet is received from the neighbor within the hold time, the BGP session will be disconnected owing to timeout. The keep-alive time is equal to or smaller than 1/3 of the hold time.

Table 11-35 Configure the keep-alive time of BGP neighbors

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-
Configure global BGP keep-alive time and hold time.	**timers bgp** *keepalive-interval holdtime-interval*	Optional. By default, the keepalive timer is 60s, the hold timer is 180s, and the session re-connection timer is 120s.

Step	Command	Description
Configure the keepalive time and hold time of a BGP neighbor or peer group.	**neighbor** { *neighbor-address* | *peer-group-name* } **timers** { *keepalive-interval holdtime-interval* | **connect** *connect-interval* }	Optional. By default, the keepalive timer is 60s, the hold timer is 180s, and the session re-connection timer is 120s.

Note:

- The keepalive time and hold time that are set for a specified neighbor have higher priorities than the global BGP keepalive time and hold time.
- Neighbors negotiate and then take the minimum hold time as the hold of the BGP session between the neighbors.
- If the keepalive time and hold time are both set to 0, the neighbor keepalive/hold function is canceled.
- If the keepalive time is longer than 1/3 of the hold time, the BGP session sends keepalive packets at the interval of 1/3 the hold time.

Configure BGP Route Detection Time

BGP mainly aims at implementing a routing process, with ASs as the routing units. Within an AS, IGP is used for routing. Therefore, BGP routes often rely on IGP routes. If the next hops or output interfaces of IGP routes that BGP relies on change, BGP detects IGP routes periodically to update BGP routes. BGP also update local BGP routes during the detection interval.

Table 11-36 Configure BGP route detection time

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-
Configure BGP route detection time.	**bgp scan-time** *time*	Optional. By default, the BGP route detection time is 60s.

Caution:

- If the BGP route detection time is set too small, BGP detect routes frequently, affecting the device performance.

Configure Quick Disconnection of EBGP Neighbors

After a BGP session is successfully set up, Keepalive messages are sent periodically between the neighbors to maintain the BGP session. If no Keepalive message or Update packet is received from the neighbor within the hold time, the BGP session will be disconnected owing to timeout. You can configure direct-connect EBGP neighbors to disconnect a BGP connection immediately after a connecting interface is down, without waiting for BGP keepalive timeout. If the EBGP neighbor quick disconnection function is cancelled, the EBGP session does not respond to an interface down event; instead, the BGP session is disconnected after timeout.

Table 11-37 Configure quick disconnection of EBGP neighbors

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-
Configure quick disconnection of EBGP neighbors.	**bgp fast-external-failover**	Optional. By default, EBGP's quick processing capability in responding to the direct-connect interface down event is enabled.

Configure the BGP Route Suppression Function

Flapping routes in a network may cause instability of the network. You can configure route attenuation to damp this type of routes so as to decrease the effect of flapping routes on the network.

A frequently flapping route will be allocated with a penalty. If the penalty exceeds the suppression threshold, the route will not be advertised to neighbors. The penalty should not be kept beyond the maximum suppression time. If no flapping occurs on the route within the half-life period, the penalty will be halved. If the penalty is lower than the threshold value, the route can be advertised to neighbors again.

- Half-life period: It is the time in which the penalty of a route is halved.
- Reuse threshold: It is the threshold for the route to resume normal use.
- Suppression threshold: It is the threshold for route suppression.
- Maximum suppression time: It is the maximum threshold value for a route penalty.

Table 11-38 Configure the BGP route suppression function

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-
Configure the BGP route attenuation period.	**bgp dampening [ibgp]** [*reach-half-life* [*reuse-value suppress-value max-suppress-time* [*unreach-half-life*]] | **route-map** *rtmap-name*]	Mandatory. By default, the route suppression function is disabled. After the function is enabled, the default route suppression half-life period is 15 minutes, the route reuse time is 750 seconds, the minimum route suppression time is 2000 seconds, the maximum route suppression time is 60 minutes, and the route penalty unreachable half-life period is 15 minutes. In the VPNv4 configuration mode, you can configure the IBGP dampening function by using the ibgp parameter.

Note:

- Route flapping not only contains addition and deletion of routes, but also contains route property changes such as next hop and MED property changes.

Configure the BGP Neighbor Refresh Capability

If the routing policy or route selection policy that is applied to a BGP neighbor changes, the route table needs to be refreshed. One way of refreshing the route table is to reset the BGP connection so as to reset the BGP session. However, this mode may result in BGP route flapping, affecting normal services. The other way is more graceful, that is, configuring the local BGP device to support the route refresh capability. If a neighbor needs to reset a route, it advertises the Route-Refresh message to the local device. After receiving the Route-Refresh message, it sends the route to the neighbor again. In this way, the route table is dynamically refreshed without the need of disconnecting the BGP session.

Table 11-39 Configure the BGP Neighbor refresh capability

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-
Enable the BGP neighbor refresh capability.	**neighbor** { *neighbor-address* | *peer-group-name* } **capability route-refresh**	Optional. By default, the BGP neighbor refresh capability is enabled.

Configure the BGP Neighbor Soft Reset Capability

If the routing policy or route selection policy that is applied to a BGP neighbor changes, the route table needs to be refreshed. One way of refreshing the route table is to reset the BGP connection so as to reset the BGP session. However, this mode may result in BGP route flapping, affecting normal services. Another way is more graceful, that is, configuring the local BGP device to support the route refresh capability. There is still another way, that is, enabling the soft reset capability of the local BGP device. By default, the BGP device reserves the routing information of each neighbor. After enabling its neighbor soft reset capability, it refreshes the neighbor routes that are kept on the local device. At this time, BGP sessions are not disconnected.

Table 11-40 Configure the BGP neighbor soft reset capability

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-
Enable the BGP neighbor soft reset capability.	**neighbor** { *neighbor-address* | *peer-group-name* } **soft-reconfiguration inbound**	Mandatory. By default, the neighbor soft reset function is disabled.

Configure the ORF Capability of BGP Neighbors

BGP implements accurate route control through abundant routing properties. It usually applies routing policies in the incoming and outgoing directions. This mode is a local BGP behavior. BGP also supports the Outbound Route Filtering (ORF) capability. It advertises the local ingress policy to its neighbors through Route-refresh packets, and then the neighbors apply the policy when they advertise routes to the local BGP device. This greatly decreases the number of exchanged route refresh packets between BGP neighbors.

To achieve successful negotiation of the ORF capability, ensure that:

- The ORF capability is enabled for both neighbors.
- "ORF send" and "ORF receive" must match. That is, if one end is "ORF send", the other end must be "ORF both" or "ORF receive". If one end is "ORF receive", the other end must be "ORF send" or "ORF both".
- The "ORF send" end must be configured with a prefix list in the incoming direction.

Table 11-41 Configure the ORF capability of BGP neighbors

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-
Apply a prefix list in the incoming direction of a neighbor.	**neighbor** { *neighbor-address* | *peer-group-name* } **prefix-list** *prefix-list-name* **in**	Mandatory. By default, no prefix list is applied to any BGP neighbor.
Configure a neighbor to support the ORF capability.	**neighbor** { *neighbor-address* | *peer-group-name* } **capability orf prefix-list** { **both** | **receive** | **send** }	Mandatory. By default, a neighbor does not support the ORF capability.

11.2.6. Configure Large-Scale BGP Network

Configuration Condition

Before configuring a large-scale BGP network, ensure that:

- BGP is enabled.
- BGP neighbors are configured and a session is set up successfully.

Configure a BGP Peer Group

A BGP peer group is a group of BGP neighbors that are configured with the same configuration policy. Any configuration that is performed on a BGP peer group will take effect on all members of the peer group. In this way, by configuring the peer group, you can perform centralized management and maintenance on the neighbors.

Table 11-42 Configure a BGP peer group

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-
Create a BGP peer group.	**neighbor** *peer-group-name* **peer-group**	Mandatory. By default, no peer group is configured, and a neighbor is not in any peer group.
Add a neighbor into the peer group.	**neighbor** *neighbor-address* **peer-group** *peer-group-name*	

Note:

- The configuration on a peer group takes effect on all members of the peer group.
- After a neighbor is added into a peer group, if some configurations of the neighbor are the same as the configurations of the peer group, the configurations of the neighbor are deleted.
- If routing policies are configured in the incoming and outgoing directions of a peer group, after the routing policies are changed, the changes do not take effect on the neighbors that have been added into the peer group. To apply the changed routing policies on the peer group members, you need to reset the peer group.

Configure a BGP Route Reflector

In a large-scale BGP network, it is required that IBGP neighbors are fully connected, that is, each BGP needs to set up connections with all IBGP neighbors. In this way, in a network which contains N BGP neighbors, the number of BGP connections is N*(N-1)/2. The larger the number of connections is, the larger the number of route advertisements is. Configuring a BGP Route Reflector (RR) is a method of reducing the number of network connections. Multiple IBGPs are categorized into a group. In this group, a BGP is specified to act as the RR, while other BGPs act as client, and BGPs that are not in the group act as non-clients. Clients set up peer relations only with the RR while they do not set up peer relations with other BGPs. This reduces the number of mandatory IBGP connections, and the number of connections is N-1.

The following shoes the routing principles of the BGP RR:

- The RR reflects the routes that it learns from non-client IBGP neighbors only to clients.
- The RR reflects the routes that it learns from clients to all clients and non-clients except the clients that initiate the routes.
- The RR reflects the routes that it learns from EBGP neighbors to all clients and non-clients.

Table 11-43 Configure a BGP route reflector

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-
Configure an RR cluster ID.	**bgp cluster-id** { *cluster-id-in-ip* | *cluster-id-in-num* }	Mandatory. By default, the route ID is used as the RR cluster ID.
Configure a neighbor as a client of the RR.	**neighbor** { *neighbor-address* | *peer-group-name* } **route-reflector-client**	Mandatory. By default, no client is specified as a client of the RR.
Configure the route reflection function between BGP neighbors.	**bgp client-to-client reflection**	Optional. By default, the route reflection function is enabled between RR clients.

Note:

- An RR cluster ID is used to identify an RR area. An RR area can contain multiple RRs, and the RRs in the RR area have the same RR cluster ID.

Configure a BGP Confederation

In a large-scale BGP network, it is required that IBGP neighbors are fully connected, that is, each BGP needs to set up connections with all IBGP neighbors. In this way, in a network which contains N BGP neighbors, the number of BGP connections is N*(N-1)/2. The larger the number of connections is, the larger the number of route advertisements is. Configuring BGP confederations is another way of reducing the number of network connections. An AS area is divided into multiple sub-AS areas, and each AS area forms a confederation. IBGP is adopted within a confederation to provide full connections, and sub-AS areas in the confederation are connected through EBGP connections. This effectively reduces the number of BGP connections.

In configuring BGP confederations, you need to assign a confederation ID for each confederation and specify members for the confederation. In the case of route reflection, only the route reflector is required to support route reflection. However, in the case of a confederation, all members in a confederation must support the confederation function.

Table 11-44 Configure a BGP confederation

Step	**Command**	**Description**
Enter the global configuration mode.	**configure terminal**	-
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-
Create a BGP confederation ID.	**bgp confederation identifier** *as-number*	Mandatory. By default, no AS number is configured for a confederation.
Configure members for the confederation.	**bgp confederation peers** *as-number-list*	Mandatory. By default, no sub-AS number is configured for a confederation.

Note:

- A confederation ID is used to identify the sub-ASs of the confederation. Confederation members are divided into the sub-ASs.

11.2.7. Configure BGP to Coordinate with BFD

Usually, there are still some intermediate devices between BGP neighbors. When an intermediate device becomes faulty, the BGP session is normal within the hold time, and the link fault caused by the intermediate device cannot be responded to in time. Bidirectional Forwarding Detection (BFD) provides a method for quickly detecting the status of a line between two devices. After BFD is enabled for BGP devices, if a line between two devices becomes faulty, BFD can quickly find the line fault and notifies BGP of the fault. It triggers BGP to quickly disconnect the session and quickly switch over to the backup line, achieving fast switchover of routes.

Configuration Condition

Before configuring BGP to coordinate with BFD, ensure that:

- BGP is enabled.
- BGP neighbors are configured and a session is set up successfully.

Configure EBGP to Coordinate with BFD

The coordination between EBGP and BFD is based on a single-hop BFD session, and BFD session parameters need to be configured in interface mode.

Table 11-45 Configure EBGP to coordinate with BFD

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-
Configure EBGP to coordinate with BFD.	**neighbor** { *neighbor-address* | *peer-group-name* } **fall-over bfd**	Mandatory. By default, the BFD function is disabled for a neighbor.
Exit the BGP configuration mode.	**exit**	-
Enter the interface configuration mode.	**interface** *interface-name*	-
Configure the minimum receive interval of a BFD session.	**bfd min-receive-interval** *milliseconds*	Optional. By default, the minimum receive interval of a BFD session is 1000ms.
Configure the minimum transmit interval of the BFD session.	**bfd min-transmit-interval** *milliseconds*	Optional. By default, the minimum transmit interval of a BFD session is 1000ms.
Configure the multiple of BFD session detection timeout.	**bfd multiplier** *number*	Optional. By default, the multiple of BFD session detection timeout is 5.

Note:

- For the related configuration of BFD, refer to the reliability technology-BFD command manual and BFD configuration manual.

Configure IBGP to Coordinate with BFD

The coordination between IBGP and BFD is based on a multi-hop BFD session, and BFD session parameters need to be configured in BGP mode.

Table 11-46 Configure IBGP to coordinate with BFD

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-
Configure IBGP to coordinate with BFD.	**neighbor** { *neighbor-address* | *peer-group-name* } **fall-over bfd**	Mandatory. By default, the BFD function is disabled for a neighbor.
Configure the minimum receive interval of the BFD session.	**bfd min-receive-interval** *milliseconds*	Optional. By default, the minimum receive interval of a BFD session is 1000ms.
Configure the minimum transmit interval of the BFD session.	**bfd min-transmit-interval** *milliseconds*	Optional. By default, the minimum transmit interval of a BFD session is 1000ms.
Configure the multiple of BFD session detection timeout.	**bfd multiplier** *number*	Optional. By default, the multiple of BFD session detection timeout is 5.

11.2.8. Configure BGP Fast Re-routing

Configuration Conditions

Before configuring BGP fast re-routing, ensure that:

- When configuring fast rerouting based on route-map, the associated route-map has been configured.
- Enable the IS-IS protocol.

Configure BGP Fast Re-routing

In the BGP network, if the link or device fails, the packet passing the fault point will be dropped or generate the loop and the caused traffic interruption will not recover until the protocol re-converges, which often lasts for several seconds. To reduce the traffic interruption time, you can configure the BGP fast re-routing. Apply the route map to set the backup next hop for the matched route. Once the active link fails, the traffic passing the faulty link will switch to the standby link at once.

Table 11-47 Configure the BGP fast re-routing

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-
Enter the BGP IPv4 unicast configuration mode	**address-family ipv4 unicast**	-
Configure fast re-routing based on route-map	**fast-reroute route-map** *route-map-name*	Mandatory By default, do not enable the fast re-routing function based on route-map.
Configure auto fast re-routing of BGP	**pic**	Mandatory By default, do not enable the auto fast re-routing function.

Caution:

- After configuring the BGP fast re-routing, you need to re-set BGP and complete the checking and backup of the initial route. Otherwise, it takes effect only for the route learned after configuration.
- For fast re-routing based on route-map, when configuring set fast-reroute backup-nexthop auto, the protocol performs auto fast re-routing.
- When using the pic mode, the protocol performs the auto fast re-routing.
- The various modes of enabling the fast re-routing are mutually exclusive.
- After configuring the BGP fast re-routing to apply the route map, set the BGP neighbor as the backup next hop via the **set fast-reroute backup-nexthop** *nexthop-address* command. If configuring the non-BGP neighbor as the backup next hop, you cannot make the fast re-routing function take effect.

11.2.9. Configure BGP LS

Configuration Conditions

Before configuring BGP LS, first complete the following task:

- Enable the OSPF protocol

- Set up the OSPF neighbor and configure distribute link-state.

Configure BGP LS

Configure BGP LS to collect the IGP protocol topology information.

Table 11-48 Configure BGP LS

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-
Configure BGP link-state address family	**address-family link-state unicast**	Optional By default, do not enable the link-state address family.
Activate the LS capability of the BGP neighbor	**neighbor** { *neighbor-address* | *peer-group-name* } **activate**	Optional By default, do not activate the LS capability of the BGP neighbor.

11.2.10. BGP Monitoring and Maintaining

Table 11-49 BGP monitoring and maintaining

Command	Description
clear ip bgp { ***** | *neighbor-address* | *as-number* | **peer-group** *peer-group-name* | **external** } [**vrf** *vrf-name* | **ipv4 unicast** | **ipv4 multicast** | **vpnv4 unicast** | **mvpn**]	Resets the BGP neighbor.
clear ip bgp [**ipv4 unicast** | **ipv4 multicast**] **dampening** [*ip-address* | *ip-address*/*mask-length*]	Clears suppressed routes.

Command	Description
clear ip bgp [**ipv4 unicast** | **ipv4 multicast**] **flap-statistics** [*ip-address* | *ip-address/mask-length*]	Clears routing flap statistics.
clear ip bgp { * | *neighbor-address* | *as-number* | **peer-group** *peer-group-name* | **external** } [**ipv4 unicast** | **ipv4 multicast** | **vpnv4 unicast** | **vrf** *vrf-name* | mvpn] { [**soft**] [**in** | **out**] }	Soft resets neighbors.
clear ip bgp { * | *neighbor-address* } **in** { **ecomm** | **prefix-filter** }	Advertises ORF to neighbors.
show ip bgp [**vpnv4** { **all** | **vrf** *vrf-name* | **rd** *route-distinguisher* }] [*ip-address* | *ip-address/mask-length*]	Display the routing information in the related BGP address family.
show ip bgp attribute-info	Display the BGP common route attributes.
show ip bgp cidr-only	Display all classful network routes of BGP.
show ip bgp community [*community-number* / *aa:nn* / **exact-match** / **local-AS** / **no-advertise** / **no-export**]	Display the routes that match the specified community property.
show ip bgp community-info	Display all community property information of BGP.
show ip bgp community-list *community-list-name*	Display the community list that is applied to routes.
show ip bgp [**vpnv4** { **all** | **vrf** *vrf-name* | **rd** *route-distinguisher* }] **dampening** { **dampened-paths** | **flap-statistics** | **parameters** }	Display the details of route attenuation.
show ip bgp filter-list *filter-list-name* [**exact-match**]	Display the routes that match the AS_PATH filter list.

Command	Description
show ip bgp inconsistent-a	Display the routes that conflict with AS_PATH.
show ip bgp ipv4 vpn-target [*vpn-rt*]	Display the VPN-TARGET route table of BGP
show ip bgp ipv4 vpn-target rt-filter [**neighbor** *ip-address*]	Display the RT filter table of the BGP neighbor
show ip bgp mvpn { **all** | **vrf** *vrf-name* | **rd** *route-distinguisher* } { **all-type** | **type** { **1** [*ip-address*] | **7**[*as:source-ip-address:group-ip-address*] } }	Display the route information in the BGP MVPN address family
show ip bgp mvpn { **all** | **vrf** *vrf-name* | **rd** *route-distinguisher* } { **neighbors** *ip-address* } { **advertised-routes** | **received-routes** | **routes** } { **all-type** | **type** { **1** [*ip-address*] | **7** [*as:source-ip-address:group-ip-address*] } }	Display the route information of the specified neighbor in the BGP MVPN address family
show ip bgp mvpn {**all** | **vrf** *vrf-name* | **rd** *route-distinguisher* | **neighbors** *ip-address* } { **all-type** | **type** { **1** | **7** } } { **statistics** }	Display the route statistics information in the BGP MVPN address family
show ip bgp [**vpnv4** { **all** | **vrf** *vrf_name* **| rd** *route-distinguisher* }] **neighbors** [*ip-address*]	Display the BGP neighbor details
show ip bgp orf ecomm	Display the ORF information of all BGP neighbors.
show ip bgp paths	Display the AS_PATH information of BGP routes.
show ip bgp prefix-list *prefix-list-name*	Display the routes that match the filter list.

Command	Description
show ip bgp quote-regexp *as-path-list-name*	Display the routes that match the AS_PATH list.
show ip bgp regexp *as-path-list-name*	Display the routes that match the AS_PATH list.
show ip bgp route-map *rtmap-name*	Display the routes that match a route map.
show ip bgp scan	Display the BGP scan information.
show ip bgp [**vpnv4** { **all** | **vrf** *vrf-name* | **rd** *route-distinguisher* } | **mvpn**] **summary**	Display the summary of BGP neighbors.

11.3. BGP Typical Configuration Example

11.3.1. Configure BGP Basic Functions

Network Requirements

- Set up EBGP neighbors between Device1 and Device2, and set up IBGP neighbors between Device2 and Device3.
- Device1 learns the interface direct route 200.0.0.0/24 of Device3, and Device3 learns the interface direct route 100.0.0.0/24 of Device1.

Network Topology

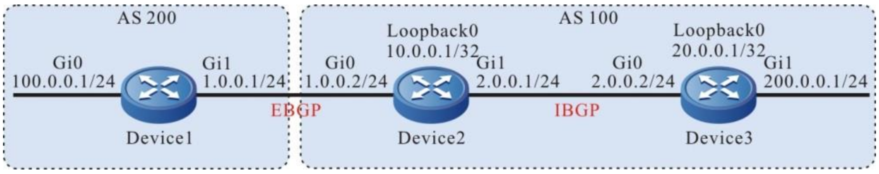

Figure 11–1 Networking for configuring basic BGP functions

Configuration Steps

Step 1: Configure the IP addresses of the interfaces. (Omitted)

Step 2: Configure OSPF so that loopback routes are reachable between devices.

#Configure Device2.

```
Device2#configure terminal
Device2(config)#router ospf 100
Device2(config-ospf)#network 10.0.0.1 0.0.0.0 area 0
Device2(config-ospf)#network 2.0.0.0 0.0.0.255 area 0
Device2(config-ospf)#exit
```

#Configure Device3.

```
Device3#configure terminal
Device3(config)#router ospf 100
Device3(config-ospf)#network 20.0.0.1 0.0.0.0 area 0
Device3(config-ospf)#network 2.0.0.0 0.0.0.255 area 0
Device3(config-ospf)#exit
```

#Query the route table of Device2.

```
Device2#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
    U - Per-user Static route
    O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

O   20.0.0.1/32 [110/2] via 2.0.0.2, 00:27:09, gigabitethernet1
```

#Query the route table of Device3.

```
Device3#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
    U - Per-user Static route
    O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

O   10.0.0.1/32 [110/2] via 2.0.0.1, 00:28:13, gigabitethernet0
```

According to the queried information, Device2 and Device3 have learnt the routes of the peer loopback interfaces by running OSPF, preparing for setting up IBGP neighbors on the loopback interfaces of Device2 and Device3.

Step 3: Configure BGP.

#Configure Device1.

Set up a direct-connect EBGP peer relation with Device2. Introduce 100.0.0.0/24 to BGP in network mode.

```
Device1#configure terminal
Device1(config)#router bgp 200
Device1(config-bgp)#neighbor 1.0.0.2 remote-as 100
Device1(config-bgp)#network 100.0.0.0 255.255.255.0
Device1(config-bgp)#exit
```

#Configure Device2.

Set up a non-direct-connect IBGP peer relation with Device3 through Loopback0, and set the next hop of the advertised route to the local device, and set up a direct-connect EBGP peer relation with Device1.

```
Device2(config)#router bgp 100
Device2(config-bgp)#neighbor 20.0.0.1 remote-as 100
```

```
Device2(config-bgp)#neighbor 20.0.0.1 update-source loopback0
Device2(config-bgp)#neighbor 1.0.0.1 remote-as 200
Device2(config-bgp)#neighbor 20.0.0.1 next-hop-self
Device2(config-bgp)#exit
```

#Configure Device3.

Set up a non-direct-connect IBGP peer relation with Device2 through Loopback0. Introduce 200.0.0.0/24 to BGP in network mode.

```
Device3(config)#router bgp 100
Device3(config-bgp)#neighbor 10.0.0.1 remote-as 100
Device3(config-bgp)#neighbor 10.0.0.1 update-source loopback0
Device3(config-bgp)#network 200.0.0.0 255.255.255.0
Device3(config-bgp)#exit
```

Note:

- To prevent route flapping, IBGP neighbors are set up through the loopback interfaces, and OSPF need to synchronize the routing information of loopback interfaces between IBGP neighbors.

Step 4: Check the result.

#On Device2, check the BGP neighbor status.

```
Device2#show ip bgp summary
BGP router identifier 10.0.0.1, local AS number 100
BGP table version is 2
2 BGP AS-PATH entries
0 BGP community entries

Neighbor       V    AS MsgRcvd MsgSent TblVer InQ OutQ Up/Down  State/PfxRcd
1.0.0.1        4   200       3       3      1   0    0 00:00:29        1
20.0.0.1       4   100       5       4      2   0    0 00:02:13        1
```

According to the numbers (Number of route prefixes received from neighbors) that are displayed in the State/PfxRcd column, BGP neighbors have been successfully set up between Device 2 and Device 1 and Device2 and Device 3.

#Query the route table of Device1.

```
Device1#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

B   200.0.0.0/24 [20/0] via 1.0.0.2, 00:15:52, gigabitethernet1
```

#Query the route table of Device2.

```
Device2#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

B   100.0.0.0/24 [20/0] via 1.0.0.1, 00:14:11, gigabitethernet0
B   200.0.0.0/24 [200/0] via 20.0.0.1, 00:17:12, gigabitethernet1
```

#Query the route table of Device3.

```
Device3#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

B   100.0.0.0/24 [200/0] via 10.0.0.1, 00:14:50, gigabitethernet0
```

Device1 has learnt the interface direct route 200.0.0.0/24 of Device3, and Device3 has learnt the interface direct route 100.0.0.0/24 of Device1.

11.3.2. Configure BGP to Redistribute Routes

Network Requirements

- Set up OSPF neighbors between Device3 and Device2, and advertise interface direct-connect route 200.0.0.0/24 to Device2.
- Set up EBGP neighbors between Device1 and Device2, and redistribute the OSPF route that Device2 learns to BGP and advertise the route to Device1.

Network Topology

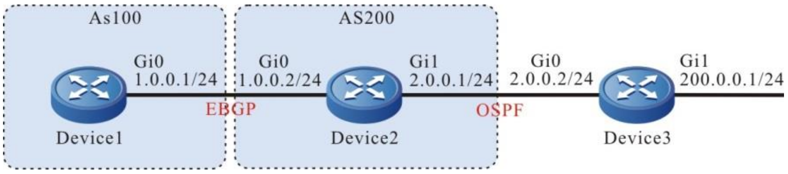

Figure 11–2 Networking for configuring BGP to redistribute routes

Configuration Steps

Step 1: Configure the IP addresses of the interfaces. (Omitted)

Step 2: Configure OSPF.

#Configure Device2.

```
Device2#configure terminal
Device2(config)#router ospf 100
Device2(config-ospf)#network 2.0.0.0 0.0.0.255 area 0
Device2(config-ospf)#exit
```

#Configure Device3.

```
Device3#configure terminal
Device3(config)#router ospf 100
Device3(config-ospf)#network 200.0.0.0 0.0.0.255 area 0
Device3(config-ospf)#network 2.0.0.0 0.0.0.255 area 0
Device3(config-ospf)#exit
```

#Query the route table of Device2.

```
Device2#show ip route ospf
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

O   200.0.0.0/24 [110/2] via 2.0.0.2, 00:01:45, gigabitethernet1
```

According to the route table, Device2 has learnt the OSPF route 200.0.0.0/24 that has been advertised by Device3.

Step 3: Configure BGP.

#Configure Device1.

```
Device1#configure terminal
Device1(config)#router bgp 200
Device1(config-bgp)#neighbor 1.0.0.2 remote-as 100
Device1(config-bgp)#exit
```

#Configure Device2.

```
Device2(config)#router bgp 100
Device2(config-bgp)#neighbor 1.0.0.1 remote-as 200
Device2(config-bgp)#exit
```

#On Device2, check the BGP neighbor status.

```
Device2#show ip bgp summary
BGP router identifier 2.0.0.1, local AS number 200
BGP table version is 2
1 BGP AS-PATH entries
0 BGP community entries

Neighbor      V    AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down  State/PfxRcd
1.0.0.1       4   100       2       2        2    0    0 00:00:42        0
```

BGP neighbors have been successfully set up between Device2 and Device1.

Step 4: Configure BGP to redistribute the OSPF route.

#Configure Device2.

```
Device2(config)#router bgp 100
Device2(config-bgp)#redistribute ospf 100
Device2(config-bgp)#exit
```

Step 5: Check the result.

#Query the BGP route table of Device2.

```
Device2#show ip bgp
BGP table version is 6, local router ID is 2.0.0.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
      Network          Next Hop          Metric LocPrf Weight Path
[0]*> 2.0.0.0/24        0.0.0.0              1        32768 ?
[0]*> 200.0.0.0/24       2.0.0.2               2        32768 ?
```

According to the queried information, OSPF routes have been successfully redistributed to BGP.

#Query the route table of Device1.

```
Device1#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external


B   2.0.0.0/24 [20/1] via 1.0.0.2, 00:06:14, gigabitethernet0
B   200.0.0.0/24 [20/2] via 1.0.0.2, 00:06:14, gigabitethernet0
```

According to the queried information, Device1 has successfully learnt routes 2.0.0.0/24 and 200.0.0.0/24.

Note:

- In an actual application, if there are two or more AS boundary routers, it is recommended that you do not redistribute routes between different routing protocols. If route redistribution must be configured, you are required to configure route control policies such as route filtering and filtration summary on the AS boundary routers to prevent routing loops.

11.3.3. Configure BGP Community Properties

Network Requirements

- Set up EBGP neighbors between Device1 and Device2.
- Device1 introduces two direct-connect routes 100.0.0.0/24 and 200.0.0.0/24 to BGP in network mode, and set different community properties for two routes that are advertised to Device2.
- When Device2 receives routes from Device1, it applies community properties in the incoming direction of a neighbor to filter route 100.0.0.0/24 and allow route 200.0.0.0/24.

Network Topology

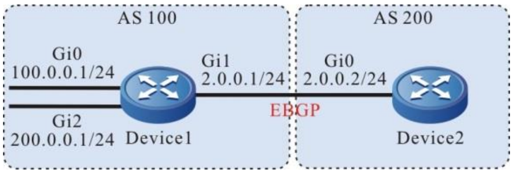

Figure 11–3 Networking for configuring BGP community properties

Configuration Steps

Step 1: Configure the IP addresses of the interfaces. (Omitted)

Step 2: Configure BGP.

#Configure Device1.

```
Device1#configure terminal
Device1(config)#router bgp 100
Device1(config-bgp)#neighbor 2.0.0.2 remote-as 200
Device1(config-bgp)#network 100.0.0.0 255.255.255.0
Device1(config-bgp)#network 200.0.0.0 255.255.255.0
Device1(config-bgp)#exit
```

#Configure Device2.

```
Device2#configure terminal
Device2(config)#router bgp 200
Device2(config-bgp)#neighbor 2.0.0.1 remote-as 100
Device2(config-bgp)#exit
```

#On Device1, check the BGP neighbor status.

```
Device1#show ip bgp summary
BGP router identifier 200.0.0.1, local AS number 100
BGP table version is 1
1 BGP AS-PATH entries
0 BGP community entries

Neighbor       V    AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down  State/PfxRcd
2.0.0.2        4   200       2       3        1    0    0 00:00:04        0
```

BGP neighbors have been successfully set up between Device1 and Device2.

#Query the route table of Device2.

```
Device2#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external
```

```
B   100.0.0.0/24 [20/0] via 2.0.0.1, 00:07:47, gigabitethernet0
B   200.0.0.0/24 [20/0] via 2.0.0.1, 00:07:47, gigabitethernet0
```

According to the queried information, Device2 has successfully learnt routes 100.0.0.0/24 and 200.0.0.0/24.

Step 3: Configure the ACL and routing policy, and set BGP community properties.

#Configure Device1.

```
Device1(config)#ip access-list standard 1
Device1(config-std-nacl)#permit 100.0.0.0 0.0.0.255
Device1(config-std-nacl)#exit
Device1(config)#ip access-list standard 2
Device1(config-std-nacl)#permit 200.0.0.0 0.0.0.255
Device1(config-std-nacl)#exit
Device1(config)#route-map CommunitySet 10
Device1(config-route-map)#match ip address 1
Device1(config-route-map)#set community 100:1
Device1(config-route-map)#exit
Device1(config)#route-map CommunitySet 20
Device1(config-route-map)#match ip address 2
Device1(config-route-map)#set community 100:2
Device1(config-route-map)#exit
```

Set different community properties for routes 100.0.0.0/24 and 200.0.0.0/24 respectively by configuring an ACL and routing policy.

Step 4: Configure a routing policy for BGP.

#Configure Device1.

```
Device1(config)#router bgp 100
Device1(config-bgp)#neighbor 2.0.0.2 route-map CommunitySet out
Device1(config-bgp)#neighbor 2.0.0.2 send-community
Device1(config-bgp)#exit
```

#Query the BGP route table of Device2.

```
Device2#show ip bgp 100.0.0.0
BGP route table entry for 100.0.0.0/24
Paths: (1 available, best #1, table Default-IP-Routing-Table)
  Not advertised to any peer
  100
    2.0.0.1 (metric 10) from 2.0.0.1 (10.0.0.1)
```

```
      Origin IGP, metric 0, localpref 100, valid, external, best
       Community: 100:1
       Last update: 00:01:06 ago

Device2#show ip bgp 200.0.0.0
BGP route table entry for 200.0.0.0/24
Paths: (1 available, best #1, table Default-IP-Routing-Table)
  Not advertised to any peer
  100
    2.0.0.1 (metric 10) from 2.0.0.1 (10.0.0.1)

      Origin IGP, metric 0, localpref 100, valid, external, best
       Community: 100:2
       Last update: 00:01:10 ago
```

According to the BGP route table of Device2, the community property of route 100.0.0.0/24 is set to 100:1, and the community properties of route 200.0.0.0/24 is set to 100:2.

Step 5: Configure BGP route filtration.

#Configure Device2.

```
Device2(config)#ip community-list 1 permit 100:2
Device2(config)#route-map communityfilter
Device2(config-route-map)#match community 1
Device2(config-route-map)#exit
Device2(config)#router bgp 200
Device2(config-bgp)#neighbor 2.0.0.1 route-map communityfilter in
Device2(config-bgp)#exit
```

Step 6: Check the result.

#Query the route table of Device2.

```
Device2#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

B   200.0.0.0/24 [20/0] via 2.0.0.1, 00:00:53, gigabitethernet0
```

According to the BGP route table of Device2, route 100.0.0.0/24 has been filtered in the incoming direction, and route 200.0.0.0/24 has been allowed.

Note:

- After a routing policy is configured on the peer, the BGP must be reset to make the configuration take effect.
- You must configure the **send-community** command to advertise the community property to the peer.

11.3.4. Configure BGP Route Reflector

Network Requirements

- Set up EBGP neighbors between Device3 and Device4, and configure Device4 to advertise route 100.0.0.0/24.
- Set up IBGP neighbors between Device2 and Device3 and between Device2 and Device1 respectively. On Device2, configure Route Reflectors (RRs), and configure Device1 and Device3 as clients, so that Device1 can learn route 100.0.0.0/24 that is advertised by Device4.

Network Topology

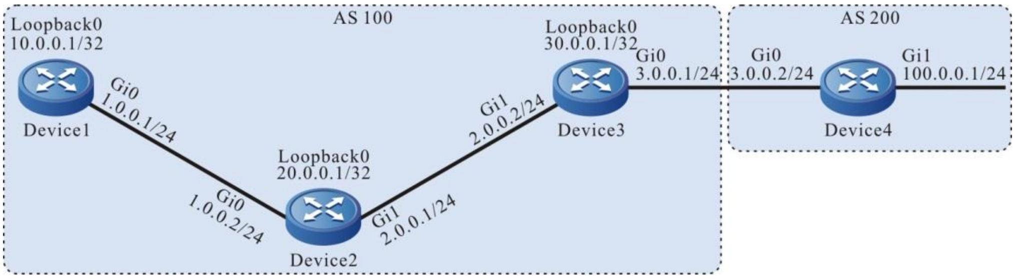

Figure 11–4 Networking for configuring a BGP route reflector

Configuration Steps

Step 1: Configure the IP addresses of the interfaces. (Omitted)

Step 2: Configure OSPF so that loopback routes are reachable between devices.

#Configure Device1.

```
Device1#configure terminal
Device1(config)#router ospf 100
Device1(config-ospf)#network 1.0.0.0 0.0.0.255 area 0
Device1(config-ospf)#network 10.0.0.1 0.0.0.0 area 0
Device1(config-ospf)#exit
```

#Configure Device2.

```
Device2#configure terminal
Device2(config)#router ospf 100
Device2(config-ospf)#network 1.0.0.0 0.0.0.255 area 0
Device2(config-ospf)#network 2.0.0.0 0.0.0.255 area 0
Device2(config-ospf)#network 20.0.0.1 0.0.0.0 area 0
Device2(config-ospf)#exit
```

#Configure Device3.

```
Device2#configure terminal
Device3(config)#router ospf 100
Device3(config-ospf)#network 2.0.0.0 0.0.0.255 area 0
Device3(config-ospf)#network 30.0.0.1 0.0.0.0 area 0
Device3(config-ospf)#exit
```

#Query the route table of Device1.

```
Device1#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

O   2.0.0.0/24 [110/2] via 1.0.0.2, 01:12:00, gigabitethernet0
O   20.0.0.1/32 [110/2] via 1.0.0.2, 01:11:47, gigabitethernet0
O   30.0.0.1/32 [110/3] via 1.0.0.2, 01:07:47, gigabitethernet0
```

#Query the route table of Device2.

```
Device2#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

O   10.0.0.1/32 [110/2] via 1.0.0.1, 01:13:02, gigabitethernet0
O   30.0.0.1/32 [110/2] via 2.0.0.2, 01:08:58, gigabitethernet1
```

#Query the route table of Device3.

```
Device3#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

O   1.0.0.0/24 [110/2] via 2.0.0.1, 01:10:04, gigabitethernet1
O   10.0.0.1/32 [110/3] via 2.0.0.1, 01:10:04, gigabitethernet1
O   20.0.0.1/32 [110/2] via 2.0.0.1, 01:10:04, gigabitethernet1
```

According to the route table, Device1, Device2, and Device3 have learnt the routes of the loopback interfaces of each other.

Step 3: Configure BGP.

#Configure Device1.

```
Device1(config)#router bgp 100
Device1(config-bgp)#neighbor 20.0.0.1 remote-as 100
```

```
Device1(config-bgp)#neighbor 20.0.0.1 update-source loopback0
Device1(config-bgp)#exit
```

#Configure Device2.

```
Device2(config)#router bgp 100
Device2(config-bgp)#neighbor 30.0.0.1 remote-as 100
Device2(config-bgp)#neighbor 30.0.0.1 update-source loopback0
Device2(config-bgp)#neighbor 10.0.0.1 remote-as 100
Device2(config-bgp)#neighbor 10.0.0.1 update-source loopback0
Device2(config-bgp)#exit
```

#Configure Device3.

```
Device3(config)#router bgp 100
Device3(config-bgp)#neighbor 3.0.0.2 remote-as 200
Device3(config-bgp)#neighbor 20.0.0.1 remote-as 100
Device3(config-bgp)#neighbor 20.0.0.1 update-source loopback0
Device3(config-bgp)#neighbor 20.0.0.1 next-hop-self
Device3(config-bgp)#exit
```

#Configure Device4.

```
Device4#configure terminal
Device4(config)#router bgp 200
Device4(config-bgp)#neighbor 3.0.0.1 remote-as 100
Device4(config-bgp)#network 100.0.0.0 255.255.255.0
Device4(config-bgp)#exit
```

#On Device2, check the BGP neighbor status.

```
Device2#show ip bgp summary
BGP router identifier 20.0.0.1, local AS number 100
BGP table version is 1
2 BGP AS-PATH entries
0 BGP community entries

Neighbor        V    AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down  State/PfxRcd
10.0.0.1        4   100       8       8        1    0    0 00:03:01        0
30.0.0.1        4   100       9       9        1    0    0 00:02:41        1
```

#On Device4, check the BGP neighbor status.

```
Device4#show ip bgp summary
BGP router identifier 100.0.0.1, local AS number 200
BGP table version is 1
1 BGP AS-PATH entries
```

```
0 BGP community entries

Neighbor        V    AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down  State/PfxRcd
3.0.0.1         4   100      19      19        1    0    0 00:05:40        0
```

According to the queried information, BGP neighbors have been set up between the devices.

#Query the BGP route table of Device3.

```
Device3#show ip bgp
BGP table version is 2, local router ID is 30.0.0.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
          S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
    Network          Next Hop           Metric LocPrf Weight Path
[B]*> 100.0.0.0/24      3.0.0.2                 0             0 200 i
```

#Query the BGP route table of Device2.

```
Device2#show ip bgp
BGP table version is 768, local router ID is 20.0.0.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
          S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
    Network          Next Hop           Metric LocPrf Weight Path
[B]*>i100.0.0.0/24      30.0.0.1                0    100      0 200 i
```

#Query the BGP route table of Device1.

```
Device1#show ip bgp
BGP table version is 2, local router ID is 10.0.0.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
          S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
    Network          Next Hop           Metric LocPrf Weight Path
```

According to the route table, Device2 and Device3 have learnt route 100.0.0.0/24, and Device2 has not advertised the route to Device1.

Step 4: Configure a BGP route reflector.

#Configure Device2.

```
Device2(config)#router bgp 100
Device2(config-bgp)#neighbor 10.0.0.1 route-reflector-client
Device2(config-bgp)#neighbor 30.0.0.1 route-reflector-client
Device2(config-bgp)#exit
```

On Device2, Device1 and Device3 have been configured as the RR clients.

Step 5: Check the result.

#Query the route table of Device1.

```
Device1#show ip bgp
BGP table version is 2, local router ID is 10.0.0.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
          S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
    Network          Next Hop         Metric LocPrf Weight Path
[B]*>i100.0.0.0/24      30.0.0.1              0    100      0 200 i

Device1#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
     U - Per-user Static route
     O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external


B   100.0.0.0/24 [200/0] via 30.0.0.1, 00:01:40, gigabitethernet0
```

On BGP of Device2, Device1 and Device3 have been configured as the RR clients, and Device2 has successfully reflects route 100.0.0.0/24 to RR client Device1.

Note:

- If you configure a peer as a RR client, the device and the neighbors of the peer will be reset.

11.3.5. Configure BGP Route Summary

Network Requirements

- Set up OSPF neighbors between Device1 and Device3, and configure Device3 to advertise routes 100.1.0.0/24 and 100.2.0.0/24 to Device1.
- Set up EBGP neighbors between Device1 and Device2.
- On Device1, aggregate routes 100.1.0.0/24 and 100.2.0.0/24 into route 100.0.0.0/14 and advertise the aggregated route to Device2.

Network Topology

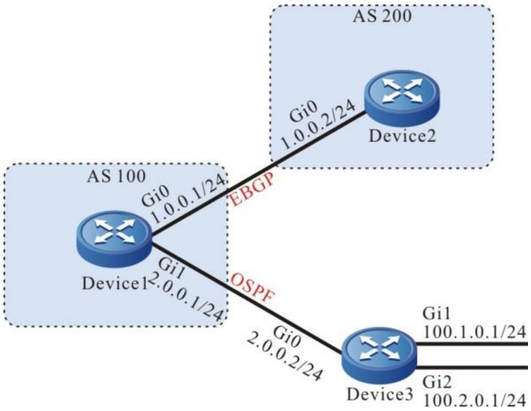

Figure 11–5 Networking for configuring BGP route summary

Configuration Steps

Step 1: Configure the IP addresses of the interfaces. (Omitted)

Step 2: Configure OSPF.

#Configure Device1.

```
Device1#configure terminal
Device1(config)#router ospf 100
Device1(config-ospf)#network 2.0.0.0 0.0.0.255 area 0
Device1(config-ospf)#exit
```

#Configure Device3.

```
Device3#configure terminal
Device3(config)#router ospf 100
Device3(config-ospf)#network 2.0.0.0 0.0.0.255 area 0
Device3(config-ospf)#network 100.1.0.0 0.0.0.255 area 0
Device3(config-ospf)#network 100.2.0.0 0.0.0.255 area 0
Device3(config-ospf)#exit
```

#Query the route table of Device1.

```
Device1#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

O   100.1.0.0/24 [110/2] via 2.0.0.2, 00:00:24, gigabitethernet1
O   100.2.0.0/24 [110/2] via 2.0.0.2, 00:00:24, gigabitethernet1
```

According to the route table, Device1 has learnt routes 100.1.0.0/24 and 100.2.0.0/24 advertised by Device3.

Step 3: Configure BGP.

#Configure Device1.

```
Device1(config)#router bgp 100
Device1(config-bgp)#neighbor 1.0.0.2 remote-as 200
Device1(config-bgp)#exit
```

#Configure Device2.

```
Device2#configure terminal
Device2(config)#router bgp 200
Device2(config-bgp)#neighbor 1.0.0.1 remote-as 100
Device2(config-bgp)#exit
```

#On Device1, check the BGP neighbor status.

```
Device1#show ip bgp summary
BGP router identifier 1.0.0.1, local AS number 100
BGP table version is 2
1 BGP AS-PATH entries
0 BGP community entries

Neighbor        V    AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down  State/PfxRcd
1.0.0.2         4   200       2       2        2    0    0 00:00:42        0
```

BGP neighbors have been successfully set up between Device1 and Device2.

Step 4: Configure BGP route summary.

Two solutions are available to satisfy network requirements.

Solution 1: Configure an aggregated static route that is targeted at null0 to introduce the aggregated static route to BGP.

#Configure Device1.

```
Device1(config)#ip route 100.0.0.0 255.252.0.0 null0
Device1(config)#router bgp 100
Device1(config-bgp)#network 100.0.0.0 255.252.0.0
Device1(config-bgp)#exit
```

Check the result.

#Query the BGP route table of Device1.

```
Device1#show ip bgp
BGP table version is 2, local router ID is 10.0.0.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
   Network          Next Hop            Metric LocPrf Weight Path
```

```
[B]*> 100.0.0.0/14       0.0.0.0                        32768 i
```

The aggregated route 100.0.0.0/14 has been generated in the BGP route table of Device1.

#Query the route table of Device2.

```
Device2#show ip bgp
BGP table version is 3, local router ID is 20.0.0.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
      Network            Next Hop           Metric LocPrf Weight Path
[B]*> 100.0.0.0/14       1.0.0.1              0            0 100 i

Device2#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

B    100.0.0.0/14 [20/0] via 1.0.0.1, 01:39:30, gigabitethernet0
```

Device2 has successfully learnt the aggregated route 100.0.0.0/14 that has been advertised by Device1.

Solution 2: First introduce common routes into BGP, and then run the **aggregate-address** command to aggregate the routes.

#Configure Device1.

```
Device1(config)#router bgp 100
Device1(config-bgp)#redistribute ospf 100
Device1(config-bgp)#aggregate-address 100.0.0.0 255.252.0.0 summary-only
Device1(config-bgp)#exit
```

Check the result.

#Query the BGP route table of Device1.

```
Device1#show ip bgp
BGP table version is 2, local router ID is 10.0.0.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
      Network            Next Hop           Metric LocPrf Weight Path
[B]*> 100.0.0.0/14       0.0.0.0                        32768 i
[B]s> 100.1.0.0/24       2.0.0.2               2         32768 i
```

```
[B]s> 100.2.0.0/24      2.0.0.2            2       32768 i
```

The aggregated route 100.0.0.0/14 has been generated in the BGP route table of Device1.

#Query the route table of Device2.

```
Device2#show ip bgp
BGP table version is 3, local router ID is 20.0.0.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
          S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
    Network          Next Hop         Metric LocPrf Weight Path
[B]*> 100.0.0.0/14      1.0.0.1            0          0 100 i

Device2#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
     U - Per-user Static route
     O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external
B   100.0.0.0/14 [20/0] via 1.0.0.1, 01:39:30, gigabitethernet0
```

Device2 has successfully learnt the aggregated route 100.0.0.0/14 that has been advertised by Device1.

Note:

- When the **aggregate-address** command is used to aggregate routes, if the extended command **summary-only** is configured, the device advertises only the aggregated route; otherwise, both common routes and aggregated routes are advertised.

11.3.6. Configure the BGP Route Selection Priority

Network Requirements

- Set up IBGP neighbors between Device1 and Device2 and between Device1 and Device3, and set up EBGP neighbors between Device4 and Device2 and between Device4 and Device3.
- Device1 advertises two routes 55.0.0.0/24 and 65.0.0.0/24 to Device4, and Device4 advertises two routes 75.0.0.0/24 and 85.0.0.0/24 to Device1.
- Modify the Local-preference property of routes on Device2 and Device3 so that Device1 selects route 75.0.0.0/24 advertised by Device2 and route 85.0.0.0/24 advertised by Device3 with priority.
- Modify the MED property of routes on Device2 and Device3 so that Device4 selects route 55.0.0.0/24 advertised by Device2 and route 65.0.0.0/24 advertised by Device3 with priority.

Network Topology

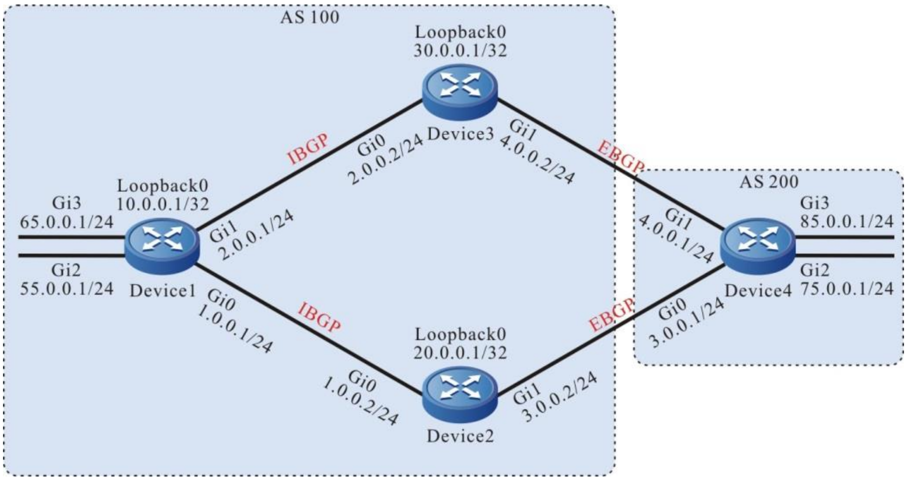

Figure 11–6 Networking for configuring the BGP route selection priority

Configuration Steps

Step 1: Configure the IP addresses of the interfaces. (Omitted)

Step 2: Configure OSPF so that loopback routes are reachable between devices.

#Configure Device1.

```
Device1#configure terminal
Device1(config)#router ospf 100
Device1(config-ospf)#network 1.0.0.0 0.0.0.255 area 0
Device1(config-ospf)#network 2.0.0.0 0.0.0.255 area 0
Device1(config-ospf)#network 10.0.0.1 0.0.0.0 area 0
Device1(config-ospf)#exit
```

#Configure Device2.

```
Device2#configure terminal
Device2(config)#router ospf 100
Device2(config-ospf)#network 1.0.0.0 0.0.0.255 area 0
Device2(config-ospf)#network 20.0.0.1 0.0.0.0 area 0
Device2(config-ospf)#exit
```

#Configure Device3.

```
Device3#configure terminal
Device3(config)#router ospf 100
Device3(config-ospf)#network 2.0.0.0 0.0.0.255 area 0
Device3(config-ospf)#network 30.0.0.1 0.0.0.0 area 0
Device3(config-ospf)#exit
```

#Query the route table of Device1.

```
Device1#show ip route
```

```
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

O   20.0.0.1/32 [110/2] via 1.0.0.2, 00:03:15, gigabitethernet0
O   30.0.0.1/32 [110/2] via 2.0.0.2, 00:01:40, gigabitethernet1
```

#Query the route table of Device2.

```
Device2#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external
O   2.0.0.0/24 [110/2] via 1.0.0.1, 00:03:54, gigabitethernet0
O   10.0.0.1/32 [110/2] via 1.0.0.1, 00:03:54, gigabitethernet0
O   30.0.0.1/32 [110/3] via 1.0.0.1, 00:02:14, gigabitethernet0
```

#Query the route table of Device3.

```
Device3#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

O   1.0.0.0/24 [110/2] via 2.0.0.1, 00:02:35, gigabitethernet0
O   10.0.0.1/32 [110/2] via 2.0.0.1, 00:02:35, gigabitethernet0
O   20.0.0.1/32 [110/3] via 2.0.0.1, 00:02:35, gigabitethernet0
```

According to the route table, Device1, Device2, and Device3 have learnt the routes of the loopback interfaces of each other.

Step 3: Configure BGP.

#Configure Device1.

```
Device1(config)#router bgp 100
Device1(config-bgp)#neighbor 20.0.0.1 remote-as 100
Device1(config-bgp)#neighbor 20.0.0.1 update-source loopback0
Device1(config-bgp)#neighbor 30.0.0.1 remote-as 100
Device1(config-bgp)#neighbor 30.0.0.1 update-source loopback0
Device1(config-bgp)#network 55.0.0.0 255.255.255.0
Device1(config-bgp)#network 65.0.0.0 255.255.255.0
Device1(config-bgp)#exit
```

#Configure Device2.

```
Device2(config)#router bgp 100
```

```
Device2(config-bgp)#neighbor 10.0.0.1 remote-as 100
Device2(config-bgp)#neighbor 10.0.0.1 update-source loopback0
Device2(config-bgp)#neighbor 10.0.0.1 next-hop-self
Device2(config-bgp)#neighbor 3.0.0.1 remote-as 200
Device2(config-bgp)#exit
```

#Configure Device3.

```
Device3(config)#router bgp 100
Device3(config-bgp)#neighbor 10.0.0.1 remote-as 100
Device3(config-bgp)#neighbor 10.0.0.1 update-source loopback0
Device3(config-bgp)#neighbor 10.0.0.1 next-hop-self
Device3(config-bgp)#neighbor 4.0.0.1 remote-as 200
Device3(config-bgp)#exit
```

#Configure Device4.

```
Device4#configure terminal
Device4(config)#router bgp 200
Device4(config-bgp)#neighbor 3.0.0.2 remote-as 100
Device4(config-bgp)#neighbor 4.0.0.2 remote-as 100
Device4(config-bgp)#network 75.0.0.0 255.255.255.0
Device4(config-bgp)#network 85.0.0.0 255.255.255.0
Device4(config-bgp)#exit
```

#On Device1, check the BGP neighbor status.

```
Device1#show ip bgp summary
BGP router identifier 10.0.0.1, local AS number 100
BGP table version is 2
2 BGP AS-PATH entries
0 BGP community entries

Neighbor       V    AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down  State/PfxRcd
20.0.0.1       4   100      11      11        2    0    0 00:07:40        2
30.0.0.1       4   100       7       7        2    0    0 00:03:59        2
```

#On Device4, check the BGP neighbor status.

```
Device4#show ip bgp summary
BGP router identifier 85.0.0.1, local AS number 200
BGP table version is 2
2 BGP AS-PATH entries
0 BGP community entries
```

```
Neighbor        V    AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down  State/PfxRcd
3.0.0.2         4   100       5       6        2    0    0 00:02:24        2
4.0.0.2         4   100       6       5        2    0    0 00:02:24        2
```

IBGP neighbors have been set up between Device1 and Device2 and between Device2 and Device3, and EBGP neighbors have been set up between Device4 and Device2 and between Device4 and Device3.

#Query the route table of Device1.

```
Device1#show ip bgp
BGP table version is 2, local router ID is 10.0.0.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
          S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
    Network          Next Hop         Metric LocPrf Weight Path
[B]*> 55.0.0.0/24      0.0.0.0              0        32768 i
[B]*> 65.0.0.0/24      0.0.0.0              0        32768 i
[B]* i75.0.0.0/24      30.0.0.1             0    100      0 200 i
[B]*>i                 20.0.0.1             0    100      0 200 i
[B]* i85.0.0.0/24      30.0.0.1             0    100      0 200 i
[B]*>i                 20.0.0.1             0    100      0 200 i


Device1#show ip route
Codes: C - connected, S - static, R - RIP,  O - OSPF, OE-OSPF External, M - Management
     D - Redirect, E - IRMP, EX - IRMP external, o - SNSP, B - BGP, i-ISIS


Gateway of last resort is not set


B   75.0.0.0/24 [200/0] via 20.0.0.1, 01:13:17, gigabitethernet0
B   85.0.0.0/24 [200/0] via 20.0.0.1, 01:13:17, gigabitethernet0
```

According to the route table, both route 75.0.0.0/24 and route 85.0.0.0/24 of Device1 select Device2 as the next-hop device.

#Query the route table of Device4.

```
Device4#show ip bgp
BGP table version is 2, local router ID is 85.0.0.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
          S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
    Network          Next Hop         Metric LocPrf Weight Path
```

```
[B]*  55.0.0.0/24       3.0.0.2            0          0 100 i
[B]*>                  4.0.0.2            0          0 100 i
[B]*  65.0.0.0/24       3.0.0.2            0          0 100 i
[B]*>                  4.0.0.2            0          0 100 i
[B]*> 75.0.0.0/24       0.0.0.0            0      32768 i
[B]*> 85.0.0.0/24       0.0.0.0            0      32768 i

Device4#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

B   55.0.0.0/24 [20/0] via 4.0.0.2, 01:25:19, gigabitethernet1
B   65.0.0.0/24 [20/0] via 4.0.0.2, 01:25:19, gigabitethernet1
```

According to the route table, both route 55.0.0.0/24 and route 65.0.0.0/24 of Device4 select Device3 as the next-hop device.

Step 4: Configure an ACL and routing policy to set local-preference and metric.

#Configure Device2.

```
Device2(config)#ip access-list standard 1
Device2(config-std-nacl)#permit 75.0.0.0 0.0.0.255
Device2(config-std-nacl)#exit
Device2(config)#ip access-list standard 2
Device2(config-std-nacl)#permit 65.0.0.0 0.0.0.255
Device2(config-std-nacl)#exit
Device2(config)#route-map SetPriority1 10
Device2(config-route-map)#match ip address 1
Device2(config-route-map)#set local-preference 110
Device2(config-route-map)#exi
Device2(config)#route-map SetPriority1 20
Device2(config-route-map)#exit
Device2(config)#route-map SetPriority2 10
Device2(config-route-map)#match ip address 2
Device2(config-route-map)#set metric 100
Device2(config-route-map)#exit
Device2(config)#route-map SetPriority2 20
Device2(config-route-map)#exit
```

On Device2, configure a routing policy to set local-preference of route 75.0.0.0/24 to 110, and set metric of route 65.0.0.0/24 to 100.

#Configure Device3.

```
Device3(config)#ip access-list standard 1
Device3(config-std-nacl)#permit 85.0.0.0 0.0.0.255
Device3(config-std-nacl)#exit
Device3(config)#ip access-list standard 2
Device3(config-std-nacl)#permit 55.0.0.0 0.0.0.255
Device3(config-std-nacl)#exit
Device3(config)#route-map SetPriority1 10
Device3(config-route-map)#match ip address 1
Device3(config-route-map)#set local-preference 110
Device3(config-route-map)#exit
Device3(config)#route-map SetPriority1 20
Device3(config-route-map)#exit
Device3(config)#route-map SetPriority2 10
Device3(config-route-map)#match ip address 2
Device3(config-route-map)#set metric 100
Device3(config-route-map)#exit
Device3(config)#route-map SetPriority2 20
Device3(config-route-map)#exit
```

On Device3, configure a routing policy to set local-preference of route 85.0.0.0/24 to 110, and set metric of route 55.0.0.0/24 to 100.

Note:

- In configuring a routing policy, you can create a filtration rule based on a prefix list or ACL. The prefix list can precisely match routing masks while the ACL cannot match routing masks.

Step 5: Configure a routing policy for BGP.

#Configure Device2.

```
Device2(config)#router bgp 100
Device2(config-bgp)#neighbor 10.0.0.1 route-map SetPriority1 out
Device2(config-bgp)#neighbor 3.0.0.1 route-map SetPriority2 out
Device2(config-bgp)#exit
```

On Device2, configure the outgoing direction of neighbor 10.0.0.1 to modify local-preference of route 75.0.0.0/24, and configure the outgoing direction of neighbor 3.0.0.1 to modify metric of route 65.0.0.0/24.

#Configure Device3.

```
Device3(config)#router bgp 100
```

```
Device3(config-bgp)#neighbor 10.0.0.1 route-map SetPriority1 out
Device3(config-bgp)#neighbor 4.0.0.1 route-map SetPriority2 out
Device3(config-bgp)#exit
```

On Device3, configure the outgoing direction of neighbor 10.0.0.1 to modify local-preference of route 85.0.0.0/24, and configure the outgoing direction of neighbor 4.0.0.1 to modify metric of route 55.0.0.0/24.

After a routing policy is configured on the peer, the BGP must be reset to make the configuration take effect.

Step 6: Check the result.

#Query the route table of Device1.

```
Device1#show ip bgp
BGP table version is 5, local router ID is 10.0.0.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
      Network          Next Hop          Metric LocPrf Weight Path
[B]*> 55.0.0.0/24        0.0.0.0              0        32768 i
[B]*> 65.0.0.0/24        0.0.0.0              0        32768 i
[B]* i75.0.0.0/24       30.0.0.1              0    100      0 200 i
[B]*>i                 20.0.0.1              0    110      0 200 i
[B]*>i85.0.0.0/24        30.0.0.1              0    110      0 200 i
[B]* i                 20.0.0.1              0    100      0 200 i

Device1#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

B   75.0.0.0/24 [200/0] via 20.0.0.1, 00:01:34, gigabitethernet0
B   85.0.0.0/24 [200/0] via 30.0.0.1, 00:00:51, gigabitethernet1
```

According to the route table, local-preference of routes 75.0.0.0/24 and 85.0.0.0/24 is modified successfully, and Device1 select route 75.0.0.0/24 that is advertised by Device2 and route 85.0.0.0/24 that is advertised by Device3 with priority.

#Query the route table of Device4.

```
Device4#show ip bgp
BGP table version is 4, local router ID is 85.0.0.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              S Stale
```

```
Origin codes: i - IGP, e - EGP, ? - incomplete
    Network          Next Hop         Metric LocPrf Weight Path
[B]*  55.0.0.0/24       4.0.0.2           100          0 100 i
[B]*>                3.0.0.2            0          0 100 i
[B]*> 65.0.0.0/24       4.0.0.2              0          0 100 i
[B]*                 3.0.0.2          100         0 100 i
[B]*> 75.0.0.0/24       0.0.0.0              0       32768 i
[B]*> 85.0.0.0/24       0.0.0.0              0       32768 i

Device4#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
     U - Per-user Static route
     O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

B   55.0.0.0/24 [20/0] via 3.0.0.2, 00:15:02, gigabitethernet0
B   65.0.0.0/24 [20/0] via 4.0.0.2, 00:14:55, gigabitethernet1
```

According to the route table, metric of routes 55.0.0.0/24 and 65.0.0.0/24 is modified successfully, and Device4 select route 55.0.0.0/24 that is advertised by Device2 and route 65.0.0.0/24 that is advertised by Device3 with priority.

Note:

- A routing policy can be used in the outgoing direction of route advertisement, and it can also be used in the incoming direction of route receiving.

11.3.7. Configure BGP Confederation

Network Requirements

- Device2, Device3, Device4, and Device5 are in the same BGP AS 200. To reduce the number of IBGP full connections, they are divided into two different ASs in one BGP confederation.
- Set up EBGP neighbors between Device1 and Device2, and advertise route 100.0.0.0/24 to AS 200.

Network Topology

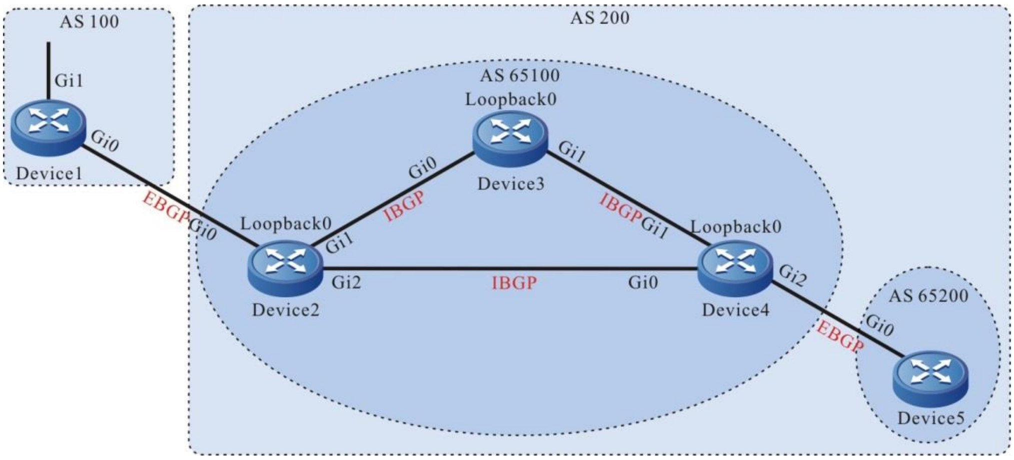

Figure 11–7 Networking for configuring a BGP confederation

Device	Interface	IP Address	Device	Interface	IP Address
Device1	Gi0	1.0.0.1/24	Device4	Gi0	3.0.0.1/24
	Gi1	100.0.0.1/24		Gi1	4.0.0.2/24
Device2	Gi0	1.0.0.2/24		Gi2	5.0.0.1/24
	Gi1	2.0.0.2/24		Loopback0	40.0.0.1/32
	Gi2	3.0.0.2/24	Device5	Gi0	5.0.0.2/24
	Loopback0	20.0.0.1/32			
Device3	Gi0	2.0.0.1/24			
	Gi1	4.0.0.1/24			
	Loopback0	30.0.0.1/32			

Configuration Steps

Step 1: Configure the IP addresses of the interfaces. (Omitted)

Step 2: Configure OSPF so that loopback routes are reachable between devices.

#Configure Device2.

```
Device2#configure terminal
Device2(config)#router ospf 100
```

```
Device2(config-ospf)#network 2.0.0.0 0.0.0.255 area 0
Device2(config-ospf)#network 3.0.0.0 0.0.0.255 area 0
Device2(config-ospf)#network 20.0.0.1 0.0.0.0 area 0
Device2(config-ospf)#exit
```

#Configure Device3.

```
Device3#configure terminal
Device3(config)#router ospf 100
Device3(config-ospf)#network 2.0.0.0 0.0.0.255 area 0
Device3(config-ospf)#network 4.0.0.0 0.0.0.255 area 0
Device3(config-ospf)#network 30.0.0.1 0.0.0.0 area 0
Device3(config-ospf)#exit
```

#Configure Device4.

```
Device4#configure terminal
Device4(config)#router ospf 100
Device4(config-ospf)#network 3.0.0.0 0.0.0.255 area 0
Device4(config-ospf)#network 4.0.0.0 0.0.0.255 area 0
Device4(config-ospf)#network 5.0.0.0 0.0.0.255 area 0
Device4(config-ospf)#network 40.0.0.1 0.0.0.0 area 0
Device4(config-ospf)#exit
```

#Configure Device5.

```
Device5#configure terminal
Device5(config)#router ospf 100
Device5(config-ospf)#network 5.0.0.0 0.0.0.255 area 0
Device5(config-ospf)#exit
```

#Query the route table of Device2.

```
Device2#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external
O   4.0.0.0/24 [110/2] via 2.0.0.1, 00:02:42, gigabitethernet1
              [110/2] via 3.0.0.1, 00:02:11, gigabitethernet2
O   30.0.0.1/32 [110/2] via 2.0.0.1, 00:02:32, gigabitethernet1
O   40.0.0.1/32 [110/2] via 3.0.0.1, 00:02:05, gigabitethernet2
```

#Query the route table of Device3.

```
Device3#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external
O   3.0.0.0/24 [110/2] via 2.0.0.2, 00:03:24, gigabitethernet0
             [110/2] via 4.0.0.2, 00:02:38, gigabitethernet1
O   20.0.0.1/32 [110/2] via 2.0.0.2, 00:03:24, gigabitethernet0
O   40.0.0.1/32 [110/2] via 4.0.0.2, 00:02:38, gigabitethernet1
```

#Query the route table of Device4.

```
Device4#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external
O   2.0.0.0/24 [110/2] via 3.0.0.2, 00:03:42, gigabitethernet0
             [110/2] via 4.0.0.1, 00:03:42, gigabitethernet1
O   20.0.0.1/32 [110/2] via 3.0.0.2, 00:03:42, gigabitethernet0
O   30.0.0.1/32 [110/2] via 4.0.0.1, 00:03:42, gigabitethernet1
```

#Query the route table of Device5.

```
Device5#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external
O   2.0.0.0/24 [110/3] via 5.0.0.1, 00:00:03, gigabitethernet0
O   3.0.0.0/24 [110/2] via 5.0.0.1, 00:00:03, gigabitethernet0
O   4.0.0.0/24 [110/2] via 5.0.0.1, 00:00:03, gigabitethernet0
O   20.0.0.1/32 [110/3] via 5.0.0.1, 00:00:03, gigabitethernet0
O   30.0.0.1/32 [110/3] via 5.0.0.1, 00:00:03, gigabitethernet0
O   40.0.0.1/32 [110/2] via 5.0.0.1, 00:00:03, gigabitethernet0
```

According to the route table, Device2, Device3, and Device4 have learnt the routes of the loopback interfaces of each other.

Step 3: Configure BGP connections in a confederation.

Configure IBGP connections in a confederation.

#Configure Device2.

```
Device2(config)#router bgp 65100
Device2(config-bgp)#neighbor 30.0.0.1 remote-as 65100
Device2(config-bgp)#neighbor 30.0.0.1 update-source loopback0
Device2(config-bgp)#neighbor 40.0.0.1 remote-as 65100
Device2(config-bgp)#neighbor 40.0.0.1 update-source loopback0
Device2(config-bgp)#neighbor 30.0.0.1 next-hop-self
Device2(config-bgp)#neighbor 40.0.0.1 next-hop-self
```

```
Device2(config-bgp)#exit
```

#Configure Device3.

```
Device3(config)#router bgp 65100
Device3(config-bgp)#neighbor 20.0.0.1 remote-as 65100
Device3(config-bgp)#neighbor 20.0.0.1 update-source loopback0
Device3(config-bgp)#neighbor 40.0.0.1 remote-as 65100
Device3(config-bgp)#neighbor 40.0.0.1 update-source loopback0
Device3(config-bgp)#exit
```

#Configure Device4.

```
Device4(config)#router bgp 65100
Device4(config-bgp)#neighbor 20.0.0.1 remote-as 65100
Device4(config-bgp)#neighbor 20.0.0.1 update-source loopback0
Device4(config-bgp)#neighbor 30.0.0.1 remote-as 65100
Device4(config-bgp)#neighbor 30.0.0.1 update-source loopback0
Device4(config-bgp)#exit
```

Configure EBGP connections in a confederation.

#Configure Device4.

```
Device4(config)#router bgp 65100
Device4(config-bgp)#neighbor 5.0.0.2 remote-as 65200
Device4(config-bgp)#exit
```

#Configure Device5.

```
Device5(config)#router bgp 65200
Device5(config-bgp)#neighbor 5.0.0.1 remote-as 65100
Device5(config-bgp)#exit
```

#On Device4, check the BGP neighbor status.

```
Device4#show ip bgp summary
BGP router identifier 40.0.0.1, local AS number 65100
BGP table version is 2
1 BGP AS-PATH entries
0 BGP community entries

Neighbor        V    AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down  State/PfxRcd
5.0.0.2         4   65200   15      15       2    0    0 00:09:40       0
20.0.0.1        4   65100   9       9        2    0    0 00:07:49       0
30.0.0.1        4   65100   7       7        2    0    0 00:05:39       0
```

IBGP neighbors have been set up between Device4 and Device2 and between Device4 and Device3, and EBGP neighbors have been set up between Device4 and Device5.

Step 4: Configure a BGP confederation.

#Configure Device1.

Configure an EBGP peer. The AS number of the peer is confederation ID 200.

```
Device1#configure terminal
Device1(config)#router bgp 100
Device1(config-bgp)#neighbor 1.0.0.2 remote-as 200
Device1(config-bgp)#network 100.0.0.0 255.255.255.0
Device1(config-bgp)#exit
```

#Configure Device2.

Configure the BGP confederation ID to 200, and configure an EBGP peer. The peer AS number is 100.

```
Device2(config)#router bgp 65100
Device2(config-bgp)#bgp confederation identifier 200
Device2(config-bgp)#neighbor 1.0.0.1 remote-as 100
Device2(config-bgp)#exit
```

#Configure Device3.

Configure the BGP confederation ID to 200.

```
Device3(config)#router bgp 65100
Device3(config-bgp)#bgp confederation identifier 200
Device3(config-bgp)#exit
```

#Configure Device4.

Configure the BGP confederation ID to 200, and configure the confederation to contain area 65100.

```
Device4#configure terminal
Device4(config)#router bgp 65100
Device4(config-bgp)#bgp confederation identifier 200
Device4(config-bgp)#bgp confederation peers 65200
Device4(config-bgp)#exit
```

#Configure Device5.

Configure the BGP confederation ID to 200, and configure the confederation to contain area 65200.

```
Device5(config)#router bgp 65200
Device5(config-bgp)#bgp confederation identifier 200
Device5(config-bgp)#bgp confederation peers 65100
Device5(config-bgp)#exit
```

Step 5: Check the result.

#On Device1, check the BGP neighbor status.

```
Device1#show ip bgp summary
BGP router identifier 100.0.0.1, local AS number 100
BGP table version is 2
1 BGP AS-PATH entries
0 BGP community entries

Neighbor       V    AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down  State/PfxRcd
1.0.0.2        4   200       6       6        2    0    0 00:02:20       0
```

EBGP neighbors have been successfully set up between Device1 and Device2.

#On Device5, query the route information.

```
Device5#show ip bgp
BGP table version is 49, local router ID is 5.0.0.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
      Network          Next Hop          Metric LocPrf Weight Path
[B]*> 100.0.0.0/24       20.0.0.1              0    100      0 (65100) 100 i

Device5#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external
B   100.0.0.0/24 [200/0] via 20.0.0.1, 00:00:38, gigabitethernet0
```

Device5 has successfully learnt route 100.0.0.0/24, and the next-hop property of the route keeps unchanged while the route is transmitted in the confederation. Device2, Device3, Device4, and Device5 belong to the same confederation, and full connections are not required. Device5 obtains external route information through Device4.

11.3.8. Configure BGP to Coordinate with BFD

Network Requirements

- Set up EBGP neighbors between Device1 and Device2 and between Device1 and Device3, and set up IBGP neighbors between Device2 and Device3.
- Device1 learns EBGP route 3.0.0.0/24 both from Device2 and Device3, and Device1 selects to forward data to the network segment 3.0.0.0/24 through Device2.
- On Device1 and Device2, configure EBGP to coordinate with BFD. When the line between Device1 and Device2 becomes faulty, BFD can quickly detect the fault and notify BGP of the fault. Then Device1 selects to forward data to network segment 3.0.0.0/24 through Device3.

Network Topology

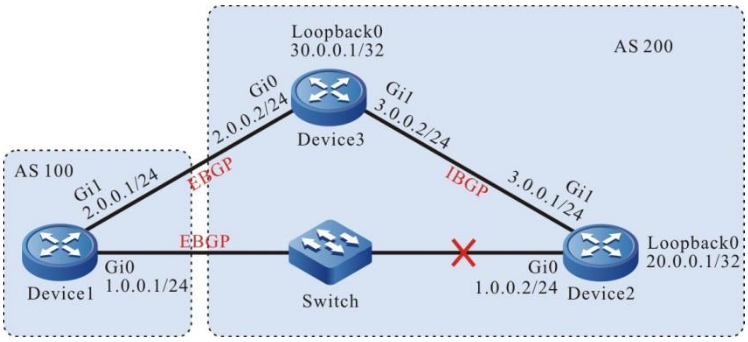

Figure 11–8 Networking for configuring BGP to coordinate with BFD

Configuration Steps

Step 1: Configure the IP addresses of the interfaces. (Omitted)

Step 2: Configure OSPF so that loopback routes are reachable between devices.

#Configure Device2.

```
Device2#configure terminal
Device2(config)#router ospf 100
Device2(config-ospf)#network 3.0.0.0 0.0.0.255 area 0
Device2(config-ospf)#network 20.0.0.1 0.0.0.0 area 0
Device2(config-ospf)#exit
```

#Configure Device3.

```
Device3#configure terminal
Device3(config)#router ospf 100
Device3(config-ospf)#network 3.0.0.0 0.0.0.255 area 0
Device3(config-ospf)#network 30.0.0.1 0.0.0.0 area 0
Device3(config-ospf)#exit
```

#Query the route table of Device2.

```
Device2#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
      U - Per-user Static route
      O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external
O   30.0.0.1/32 [110/2] via 3.0.0.2, 00:02:26, gigabitethernet1
```

#Query the route table of Device3.

```
Device3#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
      U - Per-user Static route
      O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external
O   20.0.0.1/32 [110/2] via 3.0.0.1, 00:03:38, gigabitethernet1
```

According to the route table, Device2 and Device3 have learnt the routes of the loopback interfaces of each other.

Step 3: Configure an ACL and routing policy to set metric of a route.

#Configure Device1.

```
Device1#configure terminal
Device1(config)#ip access-list standard 1
Device1(config-std-nacl)#permit 3.0.0.0 0.0.0.255
Device1(config-std-nacl)#exit
Device1(config)#route-map SetMetric
Device1(config-route-map)#match ip address 1
Device1(config-route-map)#set metric 50
Device1(config-route-map)#exit
```

The routing policy that is configured on Device1 sets the metric of route 3.0.0.0/24 to 50.

Step 4 Configure BGP, and configure Device1 with a routing policy.

#Configure Device1.

```
Device1(config)#router bgp 100
Device1(config-bgp)#neighbor 1.0.0.2 remote-as 200
Device1(config-bgp)#neighbor 2.0.0.2 remote-as 200
Device1(config-bgp)#neighbor 2.0.0.2 route-map SetMetric in
Device1(config-bgp)#exit
```

#Configure Device2.

```
Device2(config)#router bgp 200
Device2(config-bgp)#neighbor 1.0.0.1 remote-as 100
Device2(config-bgp)#neighbor 30.0.0.1 remote-as 200
Device2(config-bgp)#neighbor 30.0.0.1 update-source loopback0
Device2(config-bgp)#network 3.0.0.0 255.255.255.0
Device2(config-bgp)#exit
```

#Configure Device3.

```
Device3(config)#router bgp 200
Device3(config-bgp)#neighbor 2.0.0.1 remote-as 100
Device3(config-bgp)#neighbor 20.0.0.1 remote-as 200
Device3(config-bgp)#neighbor 20.0.0.1 update-source loopback0
Device3(config-bgp)#network 3.0.0.0 255.255.255.0
Device3(config-bgp)#exit
```

After a routing policy is configured on the peer, the BGP must be reset to make the configuration take effect.

#On Device1, check the BGP neighbor status.

```
Device1#show ip bgp summary
BGP router identifier 2.0.0.1, local AS number 100
BGP table version is 2
2 BGP AS-PATH entries
0 BGP community entries

Neighbor        V    AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down  State/PfxRcd
1.0.0.2         4   200       2       2        2    0    0 00:01:32        1
2.0.0.2         4   200       2       2        2    0    0 00:01:43        1
```

#On Device2, check the BGP neighbor status.

```
Device2#show ip bgp summary
BGP router identifier 20.0.0.1, local AS number 200
BGP table version is 2
1 BGP AS-PATH entries
0 BGP community entries

Neighbor        V    AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down  State/PfxRcd
1.0.0.1         4   100       2       2        2    0    0 00:02:52        0
30.0.0.1        4   200       3       3        2    0    0 00:02:45        1
```

BGP neighbors between Device1, Device2, and Device3 have been set up successfully.

#Query the route table of Device1.

```
Device1#show ip bgp
BGP table version is 3, local router ID is 1.0.0.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
      Network          Next Hop            Metric LocPrf Weight Path
[B]*  3.0.0.0/24       2.0.0.2                50             0 200 i
[B]*>                  1.0.0.2                 0             0 200 i

Device1#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external
B   3.0.0.0/24 [20/0] via 1.0.0.2, 00:07:19, gigabitethernet0
```

According to the route table, route 3.0.0.0/24 of Device1 selects Device2 as the next-hop device.

Step 5: Configure BGP to coordinate with BFD.

#Configure Device1.

```
Device1(config)#router bgp 100
Device1(config-bgp)#neighbor 1.0.0.2 fall-over bfd
Device1(config-bgp)#exit
Device1(config)#interface gigabitethernet0
Device1(config-if-gigabitethernet0)#bfd min-receive-interval 500
Device1(config-if-gigabitethernet0)#bfd min-transmit-interval 500
Device1(config-if-gigabitethernet0)#bfd multiplier 4
Device1(config-if-gigabitethernet0)#exit
```

#Configure Device2.

```
Device2(config)#router bgp 200
Device2(config-bgp)#neighbor 1.0.0.1 fall-over bfd
Device2(config-bgp)#exit
Device2(config)#interface gigabitethernet0
Device2(config-if-gigabitethernet0)#bfd min-receive-interval 500
Device2(config-if-gigabitethernet00)#bfd min-transmit-interval 500
Device2(config-if-gigabitethernet0)#bfd multiplier 4
Device2(config-if-gigabitethernet0)#exit
```

BFD is enabled between EBGP neighbors Device1 and Device2, and the minimum transmit interval, minimum receive interval, and detection timeout multiple of the BFD control packets have been modified.

Step 6: Check the result.

#On Device1, query the BFD session status.

```
Device1#show bfd session
OurAddr            NeighAddr          LD/RD          State        Holddown
interface
1.0.0.1          1.0.0.2          2/2        UP        2000        gigabitethernet0
```

On Device1, the BFD status is up, and the holddown time is negotiated to be 2000ms.

#If the line between Device1 and Device2 becomes faulty, the route can quickly switch over to the backup line.

#Query the route table of Device1.

```
Device1#show ip bgp
BGP table version is 6, local router ID is 1.0.0.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
        S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
```

```
    Network          Next Hop         Metric LocPrf Weight Path
[B]*> 3.0.0.0/24      2.0.0.2          50           0 200 i

Device1#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external
B   3.0.0.0/24 [20/50] via 2.0.0.2, 00:00:05, gigabitethernet1
```

The next hop of route 3.0.0.0/24 is Device3.

11.3.9. Configure BGP Fast Re-routing

Network Requirements

- All devices configure the BGP protocol.
- Device1 learns the ISIS route 192.168.1.1/32 from Device2 and Device3 at the same time. Device1 first uses the line with Device3 to forward the packet. Similarly, Device3 learns the BGP route 100.1.1.1/32 from Device1 and Device2 at the same time. Device3 first uses the line with Device1 to forward the packet.
- Device1 and Device3 enable the BGP fast re-routing. After the line between Device1 and Device3 fails, the service can switch to Device2 for communication fast.

Network Topology

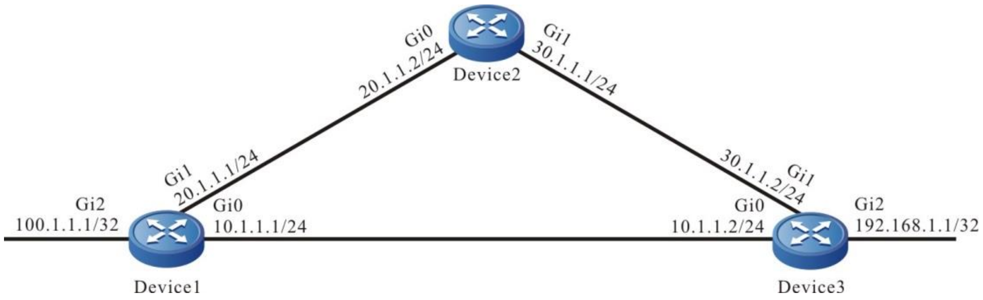

Figure 11–9 Networking for configuring BGP fast re-routing

Configuration Steps

Step 1: Configure the IP addresses of the interfaces. (Omitted)

Step 2: Configure BGP.

#Configure Device1 to set up the BGP neighbor with Device2, Device3, and the weight of the neighbor route with Device3 is 100.

```
Device1#configure terminal
Device1(config)#router bgp 100
Device1(config-bgp)#neighbor 10.1.1.2 remote-as 300
Device1(config-bgp)#neighbor 10.1.1.2 weight 100
Device1(config-bgp)#neighbor 20.1.1.2 remote-as 200
Device1(config-bgp)#network 100.1.1.1 255.255.255.255
Device1(config-bgp)#exit
```

#Configure Device2 to set up the BGP neighbor with Device1, Device3.

```
Device2#configure terminal
Device2(config)#router bgp 200
Device2(config-bgp)#neighbor 20.1.1.1 remote-as 100
Device2(config-bgp)#neighbor 30.1.1.2 remote-as 300
Device2(config-bgp)#exit
```

#Configure Device3 to set up the BGP neighbor with Device1, Device2, and the weight of the neighbor route with Device1 is 100.

```
Device3#configure terminal
Device3(config)#router bgp 300
Device3(config-bgp)#neighbor 10.1.1.1 remote-as 100
Device3(config-bgp)#neighbor 10.1.1.1 weight 100
Device3(config-bgp)#neighbor 30.1.1.1 remote-as 200
Device3(config-bgp)#network 195.168.1.1 255.255.255.255
Device3(config-bgp)#exit
```

Step 3: Configure the route policy.

#Configure Device1: configure route-map to call the ACL only matching 192.168.1.1/32, while the other network is filtered. The route matching the match rule applies the backup next-hop interface gigabitethernet1 and the next-hop address is 20.1.1.2.

```
Device1(config)#ip access-list standard 1
Device1(config-std-nacl)#permit 192.168.1.1 0.0.0.0
Device1(config-std-nacl)#exit
Device1(config)#route-map ipfrr_bgp
Device1(config-route-map)#match ip address 1
Device1(config-route-map)#set fast-reroute backup-nexthop 20.1.1.2
Device1(config-route-map)#exit
```

#Configure Device3: configure route-map to call the ACL only matching 100.1.1.1/32, while the other network is filtered. The route matching the match rule applies the backup next-hop interface gigabitethernet1 and the next-hop address is 30.1.1.1.

```
Device3(config)#ip access-list standard 1
Device3(config-std-nacl)#permit 100.1.1.1 0.0.0.0
Device3(config-std-nacl)#exit
Device3(config)#route-map ipfrr_bgp
Device3(config-route-map)#match ip address 1
Device3(config-route-map)#set fast-reroute backup-nexthop 30.1.1.1
Device3(config-route-map)#exit
```

Step 4: Configure the fast re-routing.

#Configure Device1 to enable the BGP fast re-routing.

```
Device1(config)#router bgp 100
Device1(config-bgp)#fast-reroute route-map ipfrr_bgp
Device1(config-bgp)#exit
```

#Configure Device3 to enable the BGP fast re-routing.

```
Device3(config)#router bgp 300
Device3(config-bgp)#fast-reroute route-map ipfrr_bgp
Device3(config-bgp)#exit
```

Step 5: Check the result.

#View the route table of Device1.

```
Device1#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
     U - Per-user Static route
     O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external
C   10.1.1.0/24 is directly connected, 00:45:55, gigabitethernet0
L   10.1.1.1/32 is directly connected, 00:45:55, gigabitethernet0
C   20.1.1.0/24 is directly connected, 01:29:32, gigabitethernet1
L   20.1.1.1/32 is directly connected, 01:29:32, gigabitethernet1
C   127.0.0.0/8 is directly connected, 01:36:07, lo0
L   127.0.0.1/32 is directly connected, 01:36:07, lo0
LC  100.1.1.1/32 is directly connected, 01:35:18, loopback0
B   192.168.1.1/32 [20/0] via 10.1.1.2, 00:04:40, gigabitethernet0
```

#View the fast re-route table of Device1 and you can see the route of the network 192.168.1.1/32 and the next-hop interface is gigabitethernet1.

```
Device1#show ip frr route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
     U - Per-user Static route
     O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external
B   192.168.1.1/32 [20/0] via 20.1.1.2, 00:00:02, gigabitethernet1
```

#View the backup next-hop information of Device1 and the backup interface of the fast re-routing is gigabitethernet1.

```
Device1#show nexthop frr detail
Index              : 108
Type               : FRR
```

```
Reference Count       : 1
Active Path           : master
Nexthop Address       : 10.1.1.2
Interface             : gigabitethernet0
Interface Vrf         : global
Channel ID            : 19
Link Header Length    : 18
Link Header           : 00017abc662b20120101010181000001080 0
Action                : FORWORDING
Slot                  : 0
BK Nexthop Address    : 20.1.1.2
BK Interface          : gigabitethernet1
BK Interface Vrf      : global
BK Channel ID         : 20
BK Link Header Length : 18
BK Link Header        : 00017a4554492012010101028100000208 00
BK Action             : FORWORDING
BK Slot               : 0

Total 1 entries.
```

#After the line between Device1 and Device3 fails, the system can fast detect and switch to Device2 for communication. View the route table and fast re-route table of Device1. The egress interface to the destination network 192.168.1.1/32 in the route table is switched to the backup interface gigabitethernet1.

```
Device1#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external
C   10.1.1.0/24 is directly connected, 00:45:55, gigabitethernet0
L   10.1.1.1/32 is directly connected, 00:45:55, gigabitethernet0
C   20.1.1.0/24 is directly connected, 01:29:32, gigabitethernet1
L   20.1.1.1/32 is directly connected, 01:29:32, gigabitethernet1
C   127.0.0.0/8 is directly connected, 01:36:07, lo0
L   127.0.0.1.32 is directly connected, 01:36:07, lo0
LC  100.1.1.1/32 is directly connected, 01:35:18, loopback0
B   192.168.1.1/32 [20/0] via 20.1.1.2, 00:00:40, gigabitethernet1
```

The processing mode of Device3 is similar to Device1.

11.3.10. Configure BGP-LS Basic Function

Network Requirements

- Device1 establishes an OSPF neighbor with Device2, and Device1 notifies Device2 of the loopback route.
- Device3 establishes an OSPF neighbor with Device2, and Device3 notifies Device2 of the loopback route.
- Device2 establishes a BGP-LS neighbor with Device4, and Device2 notifies Device4 of the BGP-LS route.

Network Topology

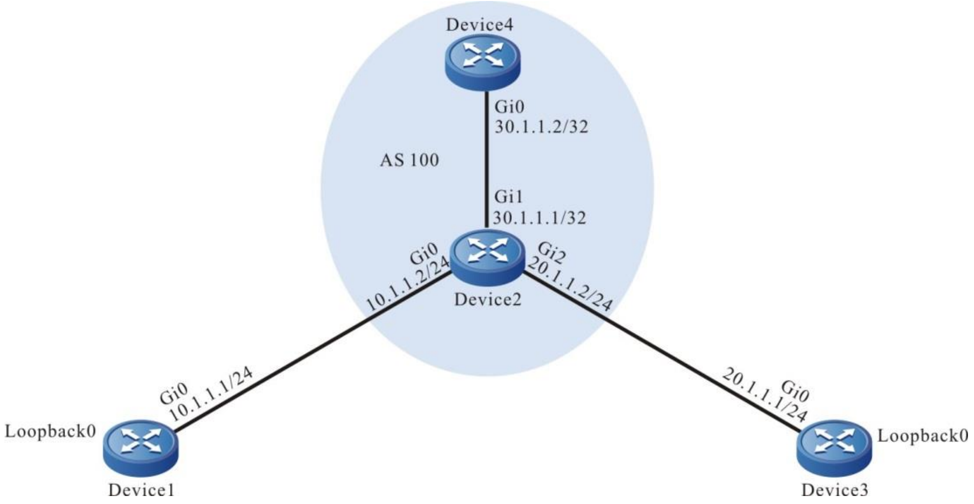

Figure 11-10 Networking of configuring BGP-LS basic functions

Configuration Steps

Step 1: Configure the IP address of the interface (omitted).

Step 2: Configure OSPF.

#Configure Device1, and Device1 and Device2 set up the OSPF neighbor of the backbone area.

```
Device1#configure terminal
Device1(config)#router ospf 65535
Device1(config-ospf)#router-id 100.1.1.1
Device1(config-ospf)#network 10.1.1.0 0.0.0.255 area 0
Device1(config-ospf)#network 1.1.1.1 0.0.0.0 area 0
Device1(config-ospf)#exit
```

#Configure Device2.

```
Device2#configure terminal
Device2(config)#router ospf 65535
Device2(config-ospf)#router-id 100.1.1.2
Device2(config-ospf)#network 10.1.1.0 0.0.0.255 area 0
Device2(config-ospf)#network 20.1.1.0 0.0.0.255 area 10
Device2(config-ospf)#exit
```

#Configure Device3, and Device3 and Device2 set up the OSPF neighbor of non-backbone area.

```
Device3#configure terminal
Device3(config)#router ospf 65535
Device3(config-ospf)#router-id 100.1.1.3
Device2(config-ospf)#network 20.1.1.0 0.0.0.255 area 10
Device3(config-ospf)#network 3.3.3.3 0.0.0.0 area 10
Device3(config-ospf)#exit
```

Step 3: Configure BGP.

#On Device2, configure BGP, and activate the BGP-LS capability of the neighbor.

```
Device2#configure terminal
Device2(config)#router bgp 65535
Device2(config-bgp)#bgp router-id 100.1.1.2
Device2(config-bgp)#neighbor 30.1.1.2 remote-as 65535
Device2(config-bgp)#address-family link-state unicast
Device2(config-bgp-af)#neighbor 30.1.1.2 activate
Device2(config-bgp-af)#exit-address-family
Device2(config-bgp)#exit
```

#On Device4, configure BGP, and activate the BGP-LS capability of the neighbor.

```
Device4#configure terminal
Device4(config)#router bgp 65535
Device4(config-bgp)#bgp router-id 100.1.1.4
Device4(config-bgp)#neighbor 30.1.1.1 remote-as 65535
Device4(config-bgp)#address-family link-state unicast
Device4(config-bgp-af)#neighbor 30.1.1.1 activate
Device4(config-bgp-af)#exit-address-family
Device4(config-bgp)#exit
```

Step 4: Configure OSPF to import the IGP topology information.

#On Device2, configure importing the IGP topology information in the OSPF address family.

```
Device2(config)#router ospf 65535
Device2(config-ospf)#distribute link-state
Device2(config-ospf)#exit
```

Step 5: Check the result.

#View the generated IGP node topology information of Device1.

```
Device2#show ip ospf 65535 link-state node
OSPF process 65535:
```

```
OSPF-LS local link state route:
Codes: N - Node route, L - Link route, P - Prefix route

 N 100.1.1.1: [area:0.0.0.0] [DR:-] flags:0x0
 N 100.1.1.2: [area:0.0.0.0] [DR:-] flags:0x10 ABR
 N 100.1.1.2: [area:0.0.0.0] [DR:10.1.1.2] flags:0x10 ABR
 N 100.1.1.2: [area:0.0.0.10] [DR:-] flags:0x10 ABR
 N 100.1.1.3: [area:0.0.0.10] [DR:-] flags:0x0
 N 100.1.1.3: [area:0.0.0.10] [DR:20.1.1.2] flags:0x0

Node route total number 6
```

#View the generated IGP link topology information of Device1.

```
Device2#show ip ospf 65535 link-state link
OSPF process 65535:
OSPF-LS local link state route:
Codes: N - Node route, L - Link route, P - Prefix route

 L [100.1.1.1:10.1.1.1][100.1.1.2:10.1.1.2][area:0.0.0.0][DR:10.1.1.2] metric:1
 L [100.1.1.2:10.1.1.2][100.1.1.1:10.1.1.1][area:0.0.0.0][DR:10.1.1.2] metric:1
 L [100.1.1.2:10.1.1.2][100.1.1.2:10.1.1.2][area:0.0.0.0][DR:10.1.1.2] metric:1
 L [100.1.1.2:10.1.1.2][100.1.1.2:10.1.1.2][area:0.0.0.0][DR:10.1.1.2] metric:1
 L [100.1.1.2:20.1.1.1][100.1.1.3:20.1.1.2][area:0.0.0.10][DR:20.1.1.2] metric:1
 L [100.1.1.3:20.1.1.2][100.1.1.2:20.1.1.1][area:0.0.0.10][DR:20.1.1.2] metric:1
 L [100.1.1.3:20.1.1.2][100.1.1.3:20.1.1.2][area:0.0.0.10][DR:20.1.1.2] metric:1
 L [100.1.1.3:20.1.1.2][100.1.1.3:20.1.1.2][area:0.0.0.10][DR:20.1.1.2] metric:1

Link route total number 8
```

#View the generated IGP prefix topology information of Device1.

```
Device2#show ip ospf 65535 link-state prefix
OSPF process 65535:
OSPF-LS local link state route:
Codes: N - Node route, L - Link route, P - Prefix route

 P 100.1.1.1:[1.1.1.1/32][area:0.0.0.0] type:1 Fwd:0.0.0.0 metric:1
 P 100.1.1.1:[10.1.1.0/24][area:0.0.0.0] type:1 Fwd:0.0.0.0 metric:1
 P 100.1.1.2:[10.1.1.0/24][area:0.0.0.0] type:1 Fwd:0.0.0.0 metric:1
 P 100.1.1.2:[20.1.1.0/24][area:0.0.0.10] type:1 Fwd:0.0.0.0 metric:1
 P 100.1.1.3:[3.3.3.3/32][area:0.0.0.10] type:1 Fwd:0.0.0.0 metric:1
 P 100.1.1.3:[20.1.1.0/24][area:0.0.0.10] type:1 Fwd:0.0.0.0 metric:1
 P 100.1.1.2:[3.3.3.3/32][area:0.0.0.0] type:2 Fwd:0.0.0.0 metric:2
 P 100.1.1.2:[20.1.1.0/24][area:0.0.0.0] type:2 Fwd:0.0.0.0 metric:1
 P 100.1.1.2:[1.1.1.1/32][area:0.0.0.10] type:2 Fwd:0.0.0.0 metric:2
 P 100.1.1.2:[10.1.1.0/24][area:0.0.0.10] type:2 Fwd:0.0.0.0 metric:1

Prefix route total number 10
```

#View the generated BGP-LS node route of Device1.

```
Device2#show bgp link-state unicast type node
BGP local router ID is 100.1.1.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
Prefix codes: E link, V node, T IP reachable route, u/U unknown,
              I Identifier, N local node, R remote node, L link, P prefix,
              L1/L2 ISIS level-1/level-2, O OSPF, D direct, S static,
              a area-ID, l link-ID, t topology-ID, s ISO-ID,
              c confed-ID/ASN, b bgp-identifier, r router-ID,
              i if-address, n nbr-address, o OSPF Route-type, p IP-prefix
              d designated router address
 Node Routes:
     Network            Next Hop          Metric    LocPrf Weight Path
[O]*> [V][O][I0x0][N[c65535][b100.1.1.2][a0.0.0.0][r100.1.1.1]]
                          0.0.0.0               0           32768 i
[O]*> [V][O][I0x0][N[c65535][b100.1.1.2][a0.0.0.0][r100.1.1.2]]
                          0.0.0.0               0           32768 i
[O]*> [V][O][I0x0][N[c65535][b100.1.1.2][a0.0.0.10][r100.1.1.2]]
                          0.0.0.0               0           32768 i
[O]*> [V][O][I0x0][N[c65535][b100.1.1.2][a0.0.0.10][r100.1.1.3]]
                          0.0.0.0               0           32768 i
[O]*> [V][O][I0x0][N[c65535][b100.1.1.2][a0.0.0.0][r100.1.1.2d10.1.1.2]]
                          0.0.0.0               0           32768 i
[O]*> [V][O][I0x0][N[c65535][b100.1.1.2][a0.0.0.10][r100.1.1.3d20.1.1.2]]
                          0.0.0.0               0           32768 i
```

#View the generated BGP-LS link route of Device1.

```
Device2#show bgp link-state unicast type link
```

```
BGP local router ID is 100.1.1.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
Prefix codes: E link, V node, T IP reachable route, u/U unknown,
              I Identifier, N local node, R remote node, L link, P prefix,
              L1/L2 ISIS level-1/level-2, O OSPF, D direct, S static,
              a area-ID, l link-ID, t topology-ID, s ISO-ID,
              c confed-ID/ASN, b bgp-identifier, r router-ID,
              i if-address, n nbr-address, o OSPF Route-type, p IP-prefix
              d designated router address
 Link Routes:
     Network          Next Hop         Metric    LocPrf Weight Path
[O]*>
[E][O][I0x0][N[c65535][b100.1.1.2][a0.0.0.0][r100.1.1.1]][R[c65535][b100.1.1.2][a0.0.0.0][r1
00.1.1.2d10.1.1.2]][L[i10.1.1.1][n10.1.1.2]]
                          0.0.0.0            0          32768 i
[O]*>
[E][O][I0x0][N[c65535][b100.1.1.2][a0.0.0.0][r100.1.1.2]][R[c65535][b100.1.1.2][a0.0.0.0][r1
00.1.1.2d10.1.1.2]][L[i10.1.1.2][n10.1.1.2]]
                          0.0.0.0            0          32768 i
[O]*>
[E][O][I0x0][N[c65535][b100.1.1.2][a0.0.0.10][r100.1.1.2]][R[c65535][b100.1.1.2][a0.0.0.10][
r100.1.1.3d20.1.1.2]][L[i20.1.1.1][n20.1.1.2]]
                          0.0.0.0            0          32768 i
[O]*>
[E][O][I0x0][N[c65535][b100.1.1.2][a0.0.0.10][r100.1.1.3]][R[c65535][b100.1.1.2][a0.0.0.10][
r100.1.1.3d20.1.1.2]][L[i20.1.1.2][n20.1.1.2]]
                          0.0.0.0            0          32768 i
[O]*>
[E][O][I0x0][N[c65535][b100.1.1.2][a0.0.0.0][r100.1.1.2d10.1.1.2]][R[c65535][b100.1.1.2][a0.
0.0.0][r100.1.1.1]][L[i10.1.1.2][n10.1.1.1]]
                          0.0.0.0            0          32768 i
[O]*>
[E][O][I0x0][N[c65535][b100.1.1.2][a0.0.0.0][r100.1.1.2d10.1.1.2]][R[c65535][b100.1.1.2][a0.
0.0.0][r100.1.1.2]][L[i10.1.1.2][n10.1.1.2]]
                          0.0.0.0            0          32768 i
[O]*>
[E][O][I0x0][N[c65535][b100.1.1.2][a0.0.0.10][r100.1.1.3d20.1.1.2]][R[c65535][b100.1.1.2][a0
.0.0.10][r100.1.1.2]][L[i20.1.1.2][n20.1.1.1]]
                          0.0.0.0            0          32768 i
```

```
[O]*>
[E][O][I0x0][N[c65535][b100.1.1.2][a0.0.0.10][r100.1.1.3d20.1.1.2]][R[c65535][b100.1.1.2][a0
.0.0.10][r100.1.1.3]][L[i20.1.1.2][n20.1.1.2]]
                 0.0.0.0            0          32768 i


Total number of prefixes 8
```

#View the generated BGP-LS prefix route of Device1.

```
Device2#show bgp link-state unicast type prefix4
BGP local router ID is 100.1.1.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
         S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
Prefix codes: E link, V node, T IP reachable route, u/U unknown,
         I Identifier, N local node, R remote node, L link, P prefix,
         L1/L2 ISIS level-1/level-2, O OSPF, D direct, S static,
         a area-ID, l link-ID, t topology-ID, s ISO-ID,
         c confed-ID/ASN, b bgp-identifier, r router-ID,
         i if-address, n nbr-address, o OSPF Route-type, p IP-prefix
         d designated router address
 Prefix4 Routes:
    Network          Next Hop         Metric     LocPrf Weight Path
[O]*> [T][O][I0x0][N[c65535][b100.1.1.2][a0.0.0.0][r100.1.1.1]][P[o0x01][p10.1.1.0/24]]
                 0.0.0.0            0          32768 i
[O]*> [T][O][I0x0][N[c65535][b100.1.1.2][a0.0.0.0][r100.1.1.1]][P[o0x01][p1.1.1.1/32]]
                 0.0.0.0            0          32768 i
[O]*> [T][O][I0x0][N[c65535][b100.1.1.2][a0.0.0.0][r100.1.1.2]][P[o0x01][p10.1.1.0/24]]
                 0.0.0.0            0          32768 i
[O]*> [T][O][I0x0][N[c65535][b100.1.1.2][a0.0.0.0][r100.1.1.2]][P[o0x02][p20.1.1.0/24]]
                 0.0.0.0            0          32768 i
[O]*> [T][O][I0x0][N[c65535][b100.1.1.2][a0.0.0.0][r100.1.1.2]][P[o0x02][p3.3.3.3/32]]
                 0.0.0.0            0          32768 i
[O]*> [T][O][I0x0][N[c65535][b100.1.1.2][a0.0.0.10][r100.1.1.2]][P[o0x01][p20.1.1.0/24]]
                 0.0.0.0            0          32768 i
[O]*> [T][O][I0x0][N[c65535][b100.1.1.2][a0.0.0.10][r100.1.1.2]][P[o0x02][p10.1.1.0/24]]
                 0.0.0.0            0          32768 i
[O]*> [T][O][I0x0][N[c65535][b100.1.1.2][a0.0.0.10][r100.1.1.2]][P[o0x02][p1.1.1.1/32]]
                 0.0.0.0            0          32768 i
[O]*> [T][O][I0x0][N[c65535][b100.1.1.2][a0.0.0.10][r100.1.1.3]][P[o0x01][p20.1.1.0/24]]
```

```
                    0.0.0.0             0           32768 i
[O]*> [T][O][I0x0][N[c65535][b100.1.1.2][a0.0.0.10][r100.1.1.3]][P[o0x01][p3.3.3.3/32]]
                    0.0.0.0             0           32768 i

Total number of prefixes 10
```

#On Device4, view the BGP-LS route received from Device2.

```
Device4#show bgp link-state unicast all-type
BGP local router ID is 100.1.1.4
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
          S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
Prefix codes: E link, V node, T IP reachable route, u/U unknown,
          I Identifier, N local node, R remote node, L link, P prefix,
          L1/L2 ISIS level-1/level-2, O OSPF, D direct, S static,
          a area-ID, l link-ID, t topology-ID, s ISO-ID,
          c confed-ID/ASN, b bgp-identifier, r router-ID,
          i if-address, n nbr-address, o OSPF Route-type, p IP-prefix
          d designated router address
 Node Routes:
    Network            Next Hop           Metric     LocPrf Weight Path
[B]*>i[V][O][I0x0][N[c65535][b100.1.1.2][a0.0.0.0][r100.1.1.1]]
                    30.1.1.1            0        100      0 i
[B]*>i[V][O][I0x0][N[c65535][b100.1.1.2][a0.0.0.0][r100.1.1.2]]
                    30.1.1.1            0        100      0 i
[B]*>i[V][O][I0x0][N[c65535][b100.1.1.2][a0.0.0.10][r100.1.1.2]]
                    30.1.1.1            0        100      0 i
[B]*>i[V][O][I0x0][N[c65535][b100.1.1.2][a0.0.0.10][r100.1.1.3]]
                    30.1.1.1            0        100      0 i
[B]*>i[V][O][I0x0][N[c65535][b100.1.1.2][a0.0.0.0][r100.1.1.2d10.1.1.2]]
                    30.1.1.1            0        100      0 i
[B]*>i[V][O][I0x0][N[c65535][b100.1.1.2][a0.0.0.10][r100.1.1.3d20.1.1.2]]
                    30.1.1.1            0        100      0 i
 Link Routes:
    Network            Next Hop           Metric     LocPrf Weight Path
[B]*>i[E][O][I0x0][N[c65535][b100.1.1.2][a0.0.0.0][r100.1.1.1]][R[c65535][b100.1.1.2][a0.0.0
.0][r100.1.1.2d10.1.1.2]][L[i10.1.1.1][n10.1.1.2]]
                    30.1.1.1            0        100      0 i
```

```
[B]*>i[E][O][I0x0][N[c65535][b100.1.1.2][a0.0.0.0][r100.1.1.2]][R[c65535][b100.1.1.2][a0.0.
0.0][r100.1.1.2d10.1.1.2]][L[i10.1.1.2][n10.1.1.2]]
                   30.1.1.1              0       100     0 i
[B]*>i[E][O][I0x0][N[c65535][b100.1.1.2][a0.0.0.10][r100.1.1.2]][R[c65535][b100.1.1.2][a0.0.
0.10][r100.1.1.3d20.1.1.2]][L[i20.1.1.1][n20.1.1.2]]
                   30.1.1.1              0       100     0 i
[B]*>i[E][O][I0x0][N[c65535][b100.1.1.2][a0.0.0.10][r100.1.1.3]][R[c65535][b100.1.1.2][a0.0.
0.10][r100.1.1.3d20.1.1.2]][L[i20.1.1.2][n20.1.1.2]]
                   30.1.1.1              0       100     0 i
[B]*>i[E][O][I0x0][N[c65535][b100.1.1.2][a0.0.0.0][r100.1.1.2d10.1.1.2]][R[c65535][b100.1.1.
2][a0.0.0.0][r100.1.1.1]][L[i10.1.1.2][n10.1.1.1]]
                   30.1.1.1              0       100     0 i
[B]*>i[E][O][I0x0][N[c65535][b100.1.1.2][a0.0.0.0][r100.1.1.2d10.1.1.2]][R[c65535][b100.1.1.
2][a0.0.0.0][r100.1.1.2]][L[i10.1.1.2][n10.1.1.2]]
                   30.1.1.1              0       100     0 i
[B]*>i[E][O][I0x0][N[c65535][b100.1.1.2][a0.0.0.10][r100.1.1.3d20.1.1.2]][R[c65535][b100.1.1
.2][a0.0.0.10][r100.1.1.2]][L[i20.1.1.2][n20.1.1.1]]
                   30.1.1.1              0       100     0 i
[B]*>i[E][O][I0x0][N[c65535][b100.1.1.2][a0.0.0.10][r100.1.1.3d20.1.1.2]][R[c65535][b100.1.1
.2][a0.0.0.10][r100.1.1.3]][L[i20.1.1.2][n20.1.1.2]]
                   30.1.1.1              0       100     0 i
 Prefix4 Routes:
     Network             Next Hop          Metric     LocPrf Weight Path
[B]*>i[T][O][I0x0][N[c65535][b100.1.1.2][a0.0.0.0][r100.1.1.1]][P[o0x01][p10.1.1.0/24]]
                   30.1.1.1              0       100     0 i
[B]*>i[T][O][I0x0][N[c65535][b100.1.1.2][a0.0.0.0][r100.1.1.1]][P[o0x01][p1.1.1.1/32]]
                   30.1.1.1              0       100     0 i
[B]*>i[T][O][I0x0][N[c65535][b100.1.1.2][a0.0.0.0][r100.1.1.2]][P[o0x01][p10.1.1.0/24]]
                   30.1.1.1              0       100     0 i
[B]*>i[T][O][I0x0][N[c65535][b100.1.1.2][a0.0.0.0][r100.1.1.2]][P[o0x02][p20.1.1.0/24]]
                   30.1.1.1              0       100     0 i
[B]*>i[T][O][I0x0][N[c65535][b100.1.1.2][a0.0.0.0][r100.1.1.2]][P[o0x02][p3.3.3.3/32]]
                   30.1.1.1              0       100     0 i
[B]*>i[T][O][I0x0][N[c65535][b100.1.1.2][a0.0.0.10][r100.1.1.2]][P[o0x01][p20.1.1.0/24]]
                   30.1.1.1              0       100     0 i
[B]*>i[T][O][I0x0][N[c65535][b100.1.1.2][a0.0.0.10][r100.1.1.2]][P[o0x02][p10.1.1.0/24]]
                   30.1.1.1              0       100     0 i
[B]*>i[T][O][I0x0][N[c65535][b100.1.1.2][a0.0.0.10][r100.1.1.2]][P[o0x02][p1.1.1.1/32]]
                   30.1.1.1              0       100     0 i
```

```
[B]*>i[T][O][I0x0][N[c65535][b100.1.1.2][a0.0.0.10][r100.1.1.3]][P[o0x01][p20.1.1.0/24]]
               30.1.1.1       0     100    0 i
[B]*>i[T][O][I0x0][N[c65535][b100.1.1.2][a0.0.0.10][r100.1.1.3]][P[o0x01][p3.3.3.3/32]]
               30.1.1.1       0     100    0 i

Total number of prefixes 2
```

12. IPV6 BGP

12.1. Overview

IPv6 BGP (BGP4 +) is extended from BGP-4. BGP-4 can only manage IPv4 routing information. To support IPv6 protocol, IETF extends BGP-4 to form IPv6 BGP. The current IPv6 BGP standard is RFC 2858 (Multiprotocol Extensions for BGP-4).

IPv6 BGP needs to reflect IPv6 network layer protocol information into NLRI (Network Layer Reachability Information) and NEXT_HOP attributes. The two NLRI attributes introduced in IPv6 BGP are:

- MP_REACH_NLRI: Multiprotocol Reachable NLRI, used to release the reachable route and next-hop information
- MP_UNREACH_NLRI: Multiprotocol Unreachable NLRI, used to cancel the unreachable route

The NEXT_HOP attribute in IPv6 BGP is represented by an IPv6 address, which can be an IPv6 global unicast address or a link local address.

IPv6 BGP uses the multi-protocol extension attributes of BGP to realize the application in IPv6 network. The original message mechanism and routing mechanism of BGP the protocol do not change.

12.2. IPv6 BGP Function Configuration

Table 12-1 IPv6 BGP function list

Configuration Tasks	
Configure an IPv6 BGP neighbor.	Configure an IBGP neighbor.
	Configure an EBGP neighbor.
	Configure a BGP passive neighbor.
	Configure an MP-BGP neighbor.
	Configure MD5, SM3 or KEYCHAIN authentication for BGP neighbors.
Configure BGP route generation.	Configure BGP to advertise local routes.
	Configuring BGP to redistribute routes.

Configuration Tasks	
Configure BGP route generation.	Configure BGP to advertise the default route.
Configure BGP route control.	Configure BGP to advertise aggregated routes.
	Configure the administrative distance of BGP routes.
	Configure routing policies in the outgoing direction of a BGP neighbor.
	Configure routing policies in the incoming direction of a BGP neighbor.
	Configure the maximum number of routes that a BGP neighbor receives.
	Configure the maximum number of BGP load balancing routes.
Configure BGP route properties.	Configure the BGP route weight.
	Configure the MED property of a BGP route.
	Configure the Local-Preference property of a BGP route.
	Configure the AS_PATH property of a BGP route.
	Configure the NEXT-HOP property of a BGP route.
	Configure the community property of a BGP route

Configuration Tasks	
Configure BGP network optimization.	Configure the keep-alive time of BGP neighbors.
	Configure BGP route detection time.
	Configure quick disconnection of EBGP neighbors.
	Configure the BGP route suppression function.
	Configure the BGP neighbor refresh capability.
	Configure the BGP neighbor soft reset capability.
	Configure the ORF capability of BGP neighbors.
Configure a large-scale BGP network.	Configure a BGP peer group.
	Configure a BGP route reflector.
	Configure a BGP confederation.
Configure BGP to coordinate with BFD.	Configure EBGP to coordinate with BFD.
	Configure IBGP to coordinate with BFD.
Configure IPv6 BGP fast re-routing	Configure IPv6 BGP fast re-routing
Configure BGP neighbor protection	Configure BGP neighbor protection

12.2.1. Configure an IPv6 BGP Neighbor

Configuration Condition

Before configuring an IPv6 BGP neighbor, ensure that:

- The link layer protocol has been configured to ensure normal communication at the link layer.
- The network layer addresses of the interfaces have been configured so that the adjacent network nodes are reachable at the network layer.

Configure an IBGP Neighbor

In configuring an IBGP neighbor, you need to set the AS of the neighbor to be the same as the AS of the local device. You can configure a Router ID for a device. The Router ID is used to uniquely identify a BGP device in setting up a BGP session. If no Router ID is configured for a device, the device selects a Router ID from interface addresses. The rules for selection are as follows:

- Select the biggest IP address from loopback interface IP addresses as the Router ID.
- If no loopback interface is configured with an IP address, select the biggest IP address from the IP addresses of other interfaces as the Router ID.
- Only when an interface is in the UP status can the IP address of the interface be elected as the Router ID.

Table 12-2 Configure an IBGP neighbor

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enable the BGP protocol and enter the BGP configuration mode.	**router bgp** *autonomous-system*	Mandatory. By default, BGP is disabled.
Configure a Router ID for the BGP device.	**bgp router-id** *router-id*	Optional. By default, the device selects a Router ID from interface addresses. The loopback interface and large IP address have the priorities.
Configure an IBGP neighbor.	**neighbor** { *neighbor-address* | *peer-group-name* } **remote-as** *as-number*	andatory. By default, no IBGP neighbor is created.

Step	Command	Description
Configure a description for an IBGP neighbor.	**neighbor** { *neighbor-address* | *peer-group-name* } **description** *description-string*	Optional. By default, no description is configured for an IBGP neighbor.
Enter the BGP IPv6 unicast configuration mode	**address-family ipv6 unicast**	-
Activate the capability of an IBGP neighbor in transmitting and receiving IPv6 unicast routes.	**neighbor** { *neighbor-address* | *peer-group-name* } **activate**	Optional. By default, the IBGP neighbor's capability in transmitting and receiving IPv6 unicast routes is not activated.

2. Configure the source address of a TCP session.

BGP uses the TCP protocol as the transport protocol. TCP features reliable transmission, ensuring that BGP protocol packets can be properly transmitted to its neighbors.

Table 12-3 Configure the source address of a TCP session

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enable the BGP protocol and enter the BGP configuration mode.	**router bgp** *autonomous-system*	-
Configure an IBGP neighbor.	**neighbor** { *neighbor-address* | *peer-group-name* } **remote-as** *as-number*	Mandatory. By default, no IBGP neighbor is created.

Step	Command	Description
Configure the source address of a TCP session of an IBGP neighbor.	**neighbor** { *neighbor-address* | *peer-group-name* } **update-source** { *interface-name* | *ipv6-address* }	Mandatory. By default, the TCP session automatically selects the address of a routing output interface as the source address.

Note:

- If there are load balancing routes, the source addresses must be configured for TCP sessions of BGP neighbors. If TCP session source addresses are not configured, if the neighbors have different optimal routes, they may use different output interfaces as their source addresses. In this way, BGP sessions may fail to set up within a period of time.

Configure an EBGP Neighbor

1. Perform basic configuration.

In configuring an EBGP neighbor, you need to set the AS of the neighbor to be different from the AS of the local device.

Table 12-4 Configure an EBGP neighbor

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enable the BGP protocol and enter the BGP configuration mode.	**router bgp** *autonomous-system*	-
Configure an EBGP neighbor.	**neighbor** { *neighbor-address* | *peer-group-name* } **remote-as** *as-number*	Mandatory. By default, no EBGP neighbor is created.

2. Configure a non-direct-connect EBGP neighbor

EBGP neighbors are located in different operation networks, and they are usually connected by a direct-connect physical link. Therefore, the default TTL value for the IP packets between EBGP neighbors is 1. In non-direct-connect operation networks, you can use a command to set the TTL value of IP packets so as to set up a BGP connection.

Table 12-5 Configure a non-direct-connect EBGP neighbor

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-
Configure an EBGP neighbor.	**neighbor** { *neighbor-address* | *peer-group-name* } **remote-as** *as-number*	Mandatory. By default, no EBGP neighbor is created.
Configure the source address of a TCP session of an EBGP neighbor.	**neighbor** { *neighbor-address* | *peer-group-name* } **update-source** { *interface-name* | *ipv6-address* }	Optional. By default, the TCP session automatically selects the address of a routing output interface as the source address.
Allow non-direct-connect EBGP neighbors to set up a connection.	**neighbor** { *neighbor-address* | *peer-group-name* } **ebgp-multihop** [*ttl-value*]	Mandatory. By default, non-direct-connect devices are not allowed to form EBGP neighbors.

Configure a BGP Passive Neighbor

In some special application environments, the BGP passive neighbor function is in need. After the passive neighbor function is enabled, the BGP does not initiate the TCP connection request for setting up a BGP neighbor relation; instead, it waits for the neighbor's connection request before setting up a neighbor relation. By default, neighbors initiate connection requests to each other. If connections conflict, they select an optimal TCP connection to form a BGP session. Before configuring a BGP passive neighbor, you need to configure a BGP neighbor.

Table 12-6 Configure a BGP passive neighbor

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-
Configure a BGP neighbor.	**neighbor** { *neighbor-address* | *peer-group-name* } **remote-as** *as-number*	Mandatory. By default, no BGP neighbor is created.
Configure a BGP passive neighbor.	**neighbor** { *neighbor-address* | *peer-group-name* } **passive**	Mandatory. By default, no passive neighbor is activated.

Configure an MP-BGP Neighbor

By default, the BGP neighbor can have the capability of receiving and sending the corresponding route when the VRF address family and VPN address family are activated. Before configuring an MP-BGP neighbor, you need to configure a BGP neighbor.

Table 12-7 Configure an MP-BGP neighbor

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-
Configure a BGP neighbor.	**neighbor** { *neighbor-address* | *peer-group-name* } **remote-as** *as-number*	Mandatory By default, do not create any BGP neighbor.
Enter the BGP IPv6 VRF configuration mode.	**address-family ipv6 vrf** *vrf-name*	-

Step	Command	Description
Configure the neighbors in the BGP IPv6 VRF address family	**neighbor** { *neighbor-address* | *peer-group-name* } **remote-as** *as-number*	Mandatory By default, do not create any BGP neighbor.
Activate the neighbors in IPv6 VRF address family.	**neighbor** { *neighbor-address* | *peer-group-name* } **activate**	Optional By default, the neighbors of the BGP IPv6 VRF configuration mode are activated.
Exit the BGP IPv6 VRF configuration mode.	**exit-address-family**	-
Enter the BGP VPNv6 configuration mode.	**address-family vpnv6** [**unicast**]	-
Activate the neighbors in BGP VPNv6 address family mode.	**neighbor** { *neighbor-address* | *peer-group-name* } **activate**	Mandatory By default, the global neighbors are not activated in the VPN address family.

Note:

- The neighbors that are configured in BGP configuration mode and BGP IPv6 unicast configuration mode are global neighbors, and the neighbors that are configured in BGP IPv6 VRF configuration mode belong only to the VRF address family.

Configure MD5, SM3 or KEYCHAIN Authentication for BGP Neighbors

BGP supports configuring MD5, SM3 or KEYCHAIN authentication to protect information exchange between neighbors. MD5, SM3 or KEYCHAIN authentication is implemented by the TCP protocol. Two neighbors must be configured with the same authentication password before a TCP connection can be set up; otherwise, if the TCP protocol fails in MD5, SM3 or KEYCHAIN authentication, the TCP connection cannot be set up. Before configuring MD5, SM3 or KEYCHAIN authentication for BGP neighbors, you need to configure BGP neighbors.

Table 12-8 Configure MD5, SM3 or KEYCHAIN authentication for BGP neighbors

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-
Configure a BGP neighbor.	**neighbor** { *neighbor-address* | *peer-group-name* } **remote-as** *as-number*	Mandatory. By default, no BGP neighbor is created.
Configure MD5, SM3 or KEYCHAIN authentication for BGP neighbors.	**neighbor** { *neighbor-address* | *peer-group-name* } { **password [sm3]** [0 | 7] *password-string* | keychain *keychain-name* }	Mandatory. By default, no MD5, SM3 or KEYCHAIN authentication is started for BGP neighbors.

12.2.2. Configure IPv6 BGP Route Generation

Configuration Condition

Before configuring IPv6 BGP route generation, ensure that:

- BGP is enabled.
- IPv6 BGP neighbors are configured and a session is set up successfully.

Configure BGP to Advertise Local Routes

BGP can use the **network** command to introduce the routes of the IPv6 route table into the BGP route table. Only when there are routes that match completely the **network** prefix and mask can the routes be introduced into the BGP route table and advertised.

In advertising a local route, you can apply a route map for the route, and you can also specify the route as the backdoor route. The backdoor route takes EBGP routes as local BGP routes and uses the administrative distance of local routes. This allows IGP routes to have higher priorities than EBGP routes. At the same time, backdoor routes will not be advertised to EBGP neighbors.

Table 12-9 Configure BGP to advertise local routes

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-
Enter the BGP IPv6 unicast configuration mode	**address-family ipv6 unicast**	-
Configure BGP to advertise local routes.	**network** *ipv6-prefix* [**route-map** *rtmap-name* [**backdoor**] | **backdoor**]	Mandatory. By default, BGP does not advertise local routes.

Note:

- The Origin property type of the local routes that are advertised by BGP is IGP.
- If you run the **network backdoor** command for an EBGP route, the administrative distance of the EBGP route changes to the local route administrative distance. (By default, the EBGP route administrative distance is 20, and the local route administrative distance is 200.), smaller than the default administrative distance of the IGP route, so that the IGP route is selected first, forming a backdoor link between EBGP neighbors.
- The route map applied to the local routes that are advertised by BGP supports match options, including as-path, community, extcommunity, ipv6 address, ipv6 nexthop, and metric, and supports the set options, including as-path, comm-list, community, extcommunity, ipv6 next-hop, local-preference, metric, origin, and weight.

Configure BGP to Redistribute Routes

BGP is not responsible for route learning. It focuses mainly on managing route properties so as to control the route direction. Therefore, BGP redistributes IGP routes to generate BGP routes and advertise the BGP routes to neighbors. When BGP redistributes IGP routes, it can apply a routing diagram.

Table 12-10 Configure BGP to redistribute routes

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-
Enter the BGP IPv6 unicast configuration mode	**address-family ipv6 unicast**	-
Configure BGP to redistribute IGP routes.	**redistribute** { **connected** | **isis** [*area-tag*] [**match** *isis-level*] | **ospf** *as-number* [**match** *route-sub-type*] | **rip** *process-id* | **static** } [**route-map** *map-name* / **metric** *value*]	Mandatory. By default, BGP does not redistribute IGP routes.

Note:

- The Origin property type of the IGP routes that are advertised by BGP is INCOMPLETE.
- The route map applied to other protocol routes that are redistributed by BGP supports match options, including as-path, community, extcommunity, ipv6 address, ipv6 nexthop, and metric, and supports the set options, including as-path, comm-list, community, extcommunity, ipv6 next-hop, local-preference, metric, origin, and weight.

Configure BGP to Advertise the Default Route

Before BGP advertises a default route to neighbors, the default route needs to be introduced. Two ways of introducing the default routes are available: Running the **neighbor default-originate** command to generate a BGP default route, and running the **default-information originate** command to redistribute the default route of another protocol.

The default route that is generated by running the **neighbor default-originate** command is route ::/0 that is automatically generated by BGP. The default route that is redistributed by running the **default-information originate** command is route 0::/0 of the redistributed protocol introduced by BGP.

Table 12-11 Configure BGP to advertise the default route

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-
Enter the BGP IPv6 unicast configuration mode	**address-family ipv6 unicast**	-
Configure BGP to generate the default route.	**neighbor** { *neighbor-address* | *peer-group-name* } **default-originate** [**route-map** *rtmap-name*]	Mandatory. By default, BGP does not generate the default route.
Configure BGP to re-distribute the default route of other protocols	**default-information originate**	Mandatory. By default, BGP does not redistribute the default route of another protocol.

Note:

- In configuring BGP to redistribute the default route of another protocol, you need to configure BGP to redistribute routes.
- In configuring BGP to generate a default route, you can apply a route map to the route.
- The route map that is applied to the default route that is generated by BGP supports set options, including as-path, comm-list, community, extcommunity, ipv6 next-hop, local-preference, metric, origin, and weigh.

12.2.3. Configure IPv6 BGP Route Control

Configuration Condition

Before configuring BGP route control, ensure that:

- BGP is enabled.
- IPv6 BGP neighbors are configured and a session is set up successfully.

Configure BGP to Advertise Aggregated Routes

In a large-scale BGP network, to decrease the number of routes that are advertised to neighbors or effectively control BGP routing, you can configure a BGP aggregated route.

Table 12-12 Configure BGP to advertise aggregated routes

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-
Enter the BGP IPv6 unicast configuration mode	**address-family ipv6 unicast**	-
Configure BGP to advertise aggregated routes.	**aggregate-address** *ipv6-prefix* [**as-set** / **summary-only** / **route-map** *rtmap-name*]	Mandatory. By default, BGP does not aggregate routes.

Note:

- When configuring BGP to advertise aggregated routes, you can specify the **summary-only** option so that BGP advertises only aggregated routes. This decreases the number of routes that are advertised.
- You can specify the **as-set** option to generate aggregation routes with the AS_PATH property.
- You can also apply a route map to the aggregation routes so as to set more abundant properties for the aggregation routes.

Configure the Administrative Distance of BGP Routes

In the IP route table, each protocol controls the administrative distance of routing. The smaller the administrative distance is, the higher the priority is .BGP affects routing by specifying the administrative distances of specified network segments. The administrative distances of the routes that cover the specified network segments will be modified. Meanwhile, ACL is applied to filter the network segments that are covered by the routes, that is, only the administrative distances of the network segment that are allowed by the ACL can be modified.

The **distance bgp** command is used to modify the management distances of external, internal, and local BGP routes. The **distance** command is only used to modify the administrative distances of specified network segments. The **distance** command has a higher priority than the **distance bgp** command. The network segments that are covered by the **distance** command use the administrative distance that is specified by the command, while the network segments that are not covered by the distance command use the administrative distance that is specified by the **distance bgp** command.

Table 12-13 Configure the administrative distance of a BGP route

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-
Enter the BGP IPv6 unicast configuration mode	**address-family ipv6 unicast**	-
Configure BGP to modify the default administrative distance.	**distance bgp** *external-distance internal-distance local-distance*	Optional. By default, the administrative distance of EBGP routes is 20, the administrative distance of IBGP routes is 200, and the administrative distance of local routes is 200.
Configure the administrative distance of a specified network segment.	**distance** *administrative-distance ipv6-prefix* [*acl-num* | *acl-name*]	

Configure Routing Policies in the Outgoing Direction of a BGP Neighbor

BGP route advertising or routing is implemented based on the powerful routing properties. When advertising routes to neighbors, you can apply routing policies to modify route properties or filter some routes. Currently, the routing policies that can be applied in the outgoing direction include:

- distribute-list: Distribution list.
- filter-list: AS_PATH property filtration list.
- prefix-list: IP prefix list.
- route-map: Route map.

Table 12-14 Configure routing policies in the outgoing direction of a BGP neighbor

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-

Step	Command	Description
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-
Enter the BGP IPv6 unicast configuration mode	**address-family ipv6 unicast**	-
Apply the distribution list in the outgoing direction.	**neighbor** { *neighbor-address* | *peer-group-name* } **distribute-list** {*access-list-num* | *access-list-name* } **out**	You can select multiple options. (However, the distribution list and the IP prefix list cannot be configured at the same time.) By default, no routing policy is configured in the outgoing direction of a BGP neighbor.
Apply the AS_PATH property filtration list in the outgoing direction.	**neighbor** { *neighbor-address* | *peer-group-name* } **filter-list** *aspath-list-name* **out**	
Apply the IP prefix list in the outgoing direction.	**neighbor** { *neighbor-address* | *peer-group-name* } **prefix-list** *prefix-list-name* **out**	
Apply a route map in the outgoing direction.	**neighbor** { *neighbor-address* | *peer-group-name* } **route-map** *rtmap-name* **out**	

Note:

- After configuring the routing policy in the outgoing direction of a BGP neighbor, you need to reset the neighbor to validate the settings.
- If you apply a route map in the outgoing direction of a route reflector, this changes only the NEXT-HOP property.
- For how to configure a filtration list, refer to the "Configure AS-PATH" section of the "Routing Policy Tools" chapter.
- In configuring routing policies in the outgoing direction of a BGP neighbor, you can configure multiple policies at the same time. BGP applies routing policies in the sequence of **distribute-list**, **filter-list**, **prefix-list**, and **route-map**. If a former policy is rejected, the latter policies will not be applied. The routing information can be advertised only after it passes all the configured policies.
- The route map that is applied in the outgoing direction of a BGP route supports match options, including as-path, community, extcommunity, ipv6 address, ip nexthop, and metric, and it supports the set options, including as-path, comm-list, community, extcommunity, ipv6 next-hop, local-preference, metric, origin, and weight.

Configure Routing Policies in the Incoming Direction of a BGP Neighbor

BGP can apply routing policies to filter received routing information or modify route properties. Similar to the policies applied in the outgoing directions, four polices are applied in the incoming directions:

- distribute-list: Distribution list.
- filter-list: AS_PATH property filtration list.
- prefix-list: IPv6 prefix list.
- route-map: Route map.

Table 12-15 Configure Routing Policies in the Incoming Direction of a BGP neighbor

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-
Enter the BGP IPv6 unicast configuration mode	**address-family ipv6 unicast**	-
Apply the distribution list in the incoming direction.	**neighbor** { *neighbor-address* | *peer-group-name* } **distribute-list** { *access-list-num* | *access-list-name* } **in**	You can select multiple options. (However, the distribution list and the IP prefix list cannot be configured at the same time.) By default, no policy is applied in the incoming direction.
Apply the AS_PATH property filtration list in the incoming direction.	**neighbor** { *neighbor-address* | *peer-group-name* } **filter-list** *aspath-list-name* **in**	
Apply the IP prefix list in the incoming direction.	**neighbor** { *neighbor-address* | *peer-group-name* } **prefix-list** *prefix-list-name* **in**	
Apply a route map in the incoming direction.	**neighbor** { *neighbor-address* | *peer-group-name* } **route-map** *rtmap-name* **in**	

Note:

- After configuring the routing policy in the incoming direction of a BGP neighbor, you need to reset the neighbor to validate the settings.

- In configuring routing policies in the incoming direction of a BGP neighbor, you can configure multiple policies at the same time. BGP applies routing policies in the sequence of **distribute-list**, **filter-list**, **prefix-list**, and **route-map**. If a former policy is rejected, the latter policies will not be applied. A route can be added into the database after it passes all the configured policies.
- The routing policies applied in the incoming direction of a BGP route support match options, including as-path, community, extcommunity, ipv6 address, ipv6 nexthop, and metric, and they support the set options, including as-path, comm-list, community, extcommunity, ipv6 next-hop, local-preference, metric, origin, and weight.

Configure Max. Routes Received by Device

The BGP device supports the limit on the number of route entries received. When the number of route entries reaches the threshold, a log alarm will be printed and the route will no longer be received. When the number of route entries is lower than the threshold, the route will be received again.

Table 12-16 Configure the maximum routes received by the BGP neighbor

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-
Enter the BGP IPv6 unicast configuration mode	**address-family ipv6 unicast**	-
Configure the maximum routes received by the device	**maximum-prefix** *{prefix-num}*	Mandatory By default, the threshold is 30000.

Configure the Maximum Number of Routes that a BGP Receives from a Neighbor

You can limit the number of routes that a BGP receives from a specified neighbor. Once the number of routes the BGP receives from the neighbor reaches a threshold, an alarm is generated or the neighbor is disconnected.

Table 12-17 Configure the maximum number of routes that a BGP Receives from a neighbor

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-

Step	Command	Description
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-
Enter the BGP IPv6 unicast configuration mode	**address-family ipv6 unicast**	-
Configure the maximum number of routes received by the neighbor.	**neighbor** { *neighbor-address* | *peer-group-name* } **maximum-prefix** *prefix-num* [*threshold-value*] [**warning-only**]	Mandatory. By default, the number of routes received by the neighbor.

Note:

- If the **warning-only** option is not specified, after the number of routes that the BGP receives from the neighbor reaches the maximum number, the BGP session is automatically disconnected.
- If the **warning-only** option is specified, after the number of routes that the BGP receives from the neighbor reaches the maximum number, a warning message is displayed, but route learning continues.

Configure the Maximum Number of BGP Load Balancing Routes

In a BGP networking environment, if several paths with the same cost are available to reach the same destination, you can configure the number of BGP load balancing routes for load balancing.

Table 12-18 Configure the Maximum number of BGP load balancing routes

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-
Enter the BGP IPv6 unicast configuration mode	**address-family ipv6 unicast**	-

Step	Command	Description
Configure the maximum number of IBGP load balancing routes.	**maximum-paths ibgp** *number*	Mandatory. By default, IBGP does not support load balancing routes.
Configure the maximum number of EBGP load balancing routes.	**maximum-paths** *number*	Mandatory. By default, EBGP does not support load balancing routes.
Configure the maximum of IBGP and EBGP load balancing routes	**maximum-paths eibgp** *number*	Mandatory By default, IBGP and EBGP route cannot perform load balancing routing at the same time.

Note:

- After the maximum number of EBGP load balancing routes is configured, load balancing takes effect only when EBGP routes are selected with priority.
- In different BGP configuration modes, the commands for configuring the maximum number of load balancing routes are different. For details, refer to the description of **maximum-paths** in the BGP technical manual.

12.2.4. Configure IPv6 BGP Route Properties

Configuration Condition

Before configuring BGP route properties, ensure that:

- BGP is enabled.
- IPv6 BGP neighbors are configured and a session is set up successfully.

Configure the BGP Route Weight

In BGP routing, the first rule is to compare the weights of routes. The larger the weight of a route is, the higher the priority it has. The weight of a route is the local property of the device, and it cannot be transferred to other BGP neighbors. The value range of a route weight is 1-65535. By default, the weight of a route that has been learnt from a neighbor is 0, and the weights of all routes that are generated by the local device are all 32768.

Table 12-19 Configure the BGP route weight

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-
Configure the weight of a route of a neighbor or peer group.	**neighbor** { *neighbor-address* | *peer-group-name* } **weight** *weight-num*	Mandatory. By default, the weight of a route of a neighbor is 0.

Configure the MED Property of a BGP Route

Multi-Exit Discriminator (MED) properties are used to select the optimal route for the traffic that enters an AS. If the other routing conditions are the same and BGP learns several routes with the same destination from different EBGP neighbors, BGP select the route with the minimum MED value as the optimal ingress.

MED sometimes is also called external metric. It is marked as a "metric" in the BGP route table. BGP advertises the MED properties of the routes that it has learnt from neighbors to IBGP neighbors, but BGP does not advertise the MED properties to EBGP neighbors. Therefore, MED properties are applicable to only adjacent ASs.

1. Configure BGP to allow comparing MEDs of neighbor routes from different ASs.

By default, BGP implements MED route selection only among the routes that are from the same AS. However, you can run the **bgp always-compare-med** command to let BGP ignore the limitation on the same AS in MED route selection.

Table 12-20 Configure BGP to allow comparing MEDs of neighbor routes from different ASs

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-

Step	Command	Description
Configure BGP to allow comparing MEDs of neighbor routes from different ASs.	**bgp always-compare-med**	Mandatory. By default, BGP allows only comparing MEDs of neighbor routes from the same AS.

2. Configure BGP to sort and select MEDs according to AS_PATH groups.

By default, BGP is not enabled to sort and select MEDs according to route AS_PATH groups. To enable the function, run the **bgp deterministic-med** command. In route selection, all routes are organized based on AS_PATHs. In each AS_PATH group, routes are sorted based MED values. The route with the minimum MED value is selected as the optimal route in the group.

Table 12-21 Configure BGP to sort and select MEDs according to AS_PATH groups

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-
Compare confederation MEDs in BGP route selection.	**bgp deterministic-med**	Mandatory. By default, confederation MEDs are not compared in BGP route selection.

3. Configure to compare MEDs of routes in the local confederation.

By default, the MED values of EBGP routes from different ASs are not compared. The setting is valid for the EBGP routes of confederations. To enable comparison of MED values of routes of the local confederation, run the **bgp bestpath med confed** command.

Table 12-22 Configure BGP to compare MEDs of routes in the local confederation

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-
Configure BGP to compare MED values of routes in the local confederation.	**bgp bestpath med confed**	Mandatory. By default, the MED values of routes in the local confederations will not be compared.

4. Configure a route map to modify MED properties.

In transmitting and receiving routes, you can apply a route map to modify MED properties.

Table 12-23 Configure a route map to modify MED properties

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-
Enter the BGP IPv6 unicast configuration mode	**address-family ipv6 unicast**	-
Configuring a route map to modify MED properties.	**neighbor** { *neighbor-address* | *peer-group-name* } **route-map** *rtmap-name* **in** | **out**	Mandatory. By default, no route map is applied to any neighbor.

Note:

- In configuring a route map to modify an MED property, you can use the **set metric** command to modify the MED property. For details, refer to Routing Policy Tools-Command Manual-**set metric**.

- After the **neighbor attribute-unchanged** command is configured, the MED properties of neighbors cannot be changed by the route map that is applied.

Configure the Local-Preference Property of a BGP Route

Local-Preference properties are transferred only between IBGP neighbors. Local-Preference is used to select the optimal egress of an AS. The route with the maximum Local-Preference will be selected with priority.

The value range of Local-Preference is 0-4294967295. The larger the value is, the higher priority the route has. By default, the Local-Preference value of all the routes that are advertised to IBGP neighbors is 100. You can use the **bgp default local-preference** command or the route map to modify the Local-Preference property value.

1. Configure BGP to modify the default Local-Preference property.

Table 12-24 Configure BGP to modify the default local-preference property

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-
Configure the default value of BGP Local-Preference property.	**bgp default local-preference** *local-value*	Optional. By default, the Local-Preference value is 100.

2. Configure the route map to modify the Local-Preference property.

Table 12-25 Configure the route map to modify the local-preference property

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-
Enter the BGP IPv6 unicast configuration mode	**address-family ipv6 unicast**	-

Step	Command	Description
Configure the route map to modify the Local-Preference property.	**neighbor** { *neighbor-address* | *peer-group-name* } **route-map** *rtmap-name* **in** | **out**	Mandatory. By default, the route map is not applied to any neighbor.

Note:

- In configuring a route map to modify the Local-Preference property, you can use the **set local-preference** command to modify the Local-Preference property. For details, refer to Routing Policy Tools-Command Manual-set local-preference.

Configure the AS_PATH Property of a BGP Route

1. Configure BGP to ignore AS_PATHs in route selection.

If the other conditions are the same, BGP selects the route with the shortest AS-PATH in route selection. To cancel route selection based on AS_PATHs, run the **bgp bestpath as-path ignore** command.

Table 12-26 Configure BGP to ignore AS_PATHs in route selection

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-
Configure BGP to ignore AS_PATHs in route selection.	**bgp bestpath as-path ignore**	Mandatory. By default, the AS_PATH values are compared in route selection.

2. Configure the number of local ASs that BGP allows to repeat.

To prevent routing loops, BGP checks the AS_PATH properties of the routes that are received from neighbors, and the routes containing the local AS number will be discarded. However, you can run the **neighbor allowas-in** command to allow the AS_PATH properties of the routes that the BGP receives to contain the local AS number, and you can configure the number of ASs that can be contained.

Table 12-27 Configure the number of local ASs that BGP allows to repeat

Step	**Command**	**Description**
Enter the global configuration mode.	**configure terminal**	-
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-
Enter the BGP IPv6 unicast configuration mode	**address-family ipv6 unicast**	-
Configure the number of ASs that are allowed to repeat.	**neighbor** { *neighbor-address* | *peer-group-name* } **allowas-in** [*as-num*]	Mandatory. By default, the AS_PATH properties of the routes that are received from neighbors do not allow the local AS number.

3. Configure BGP to remove the private AS number when advertising routes to neighbors.

In a large-scale BGP network, the AS_PATH properties of routes contain federation or community property. By default, BGP provides the private AS properties when it advertises routes to neighbors. To mask private network information, run the **neighbor remove-private-AS** command to remove the private AS number.

Table 12-28 Configure BGP to Remove the Private AS number when advertising routes to neighbors

Step	**Command**	**Description**
Enter the global configuration mode.	**configure terminal**	-
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-
Enter the BGP IPv6 unicast configuration mode	**address-family ipv6 unicast**	-

Step	Command	Description
Configure BGP to remove the private AS number when advertising routes to neighbors.	**neighbor** { *neighbor-address* | *peer-group-name* } **remove-private-AS**	Mandatory. By default, when BGP advertise routes to neighbors, it provides the private AS number.

4. Configure to check the validity of the first AS number of an EBGP route.

When BGP advertises a route to EBGP neighbors, it compresses the local AS number to the starting position of the AS_PATH, and the AS that advertises the route first is located at the end. Usually, the first AS of a route that EBGP receives must be the same as the neighbor AS number; otherwise, the route will be discarded.

Table 12-29 Configure to check the validity of the first AS number of an EBGP route

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-
Configure to check the validity of the first AS number of an EBGP route.	**bgp enforce-first-as**	Mandatory. By default, BGP does not enable the mechanism for checking the first AS number.

5. Configure a route map to modify AS_PATH properties.

BGP supports configuring a route map to modify AS_PATH properties. You can run the **set as-path prepend** command to add more routing properties, or use the **set as-path replace** command to replace the AS_PATH properties so as to affect neighbor routing. In using the **set as-path prepend** function, first use the local AS to add AS_PATH. If you use another AS, the AS must be emphasized to prevent the AS from rejecting routes that are advertised to it.

Table 12-30 Configure a Route map to modify AS_PATH properties

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-
Enter the BGP IPv6 unicast configuration mode	**address-family ipv6 unicast**	-
Configure a route map to modify AS_PATH properties.	**neighbor** { *neighbor-address* | *peer-group-name* } **route-map** *rtmap-name* **in | out**	Mandatory. By default, no route map is applied to any neighbor.

Note:

- In configuring a route map to modify an AS_PATH property, you can use the **set as-path prepend** command or the **set as-path replace** command to modify the AS_PATH property. For details, refer to Routing Policy Tools-Command Manual-**set as-path**.

Configure the NEXT-HOP Property of a BGP Route

When BGP advertises routes to IBGP neighbors, it does not change the routing properties (including the NEXT-HOP property). When BGP advertises the routes that are learned from EBGP neighbors to IBGP neighbors, you can run the **neighbor next-hop-self** command to modify the next-hop property of the routes advertised to BGP neighbors to the local IP address. You can apply a route map to modify the next hop property.

1. Configure BGP to use the local IP address as the next hop of a route.

Table 12-31 Configure BGP to use the local IP address as the next hop of a route

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-

Step	Command	Description
Enter the BGP IPv6 unicast configuration mode	**address-family ipv6 unicast**	-
Configure BGP to use the local IP address as the next hop when advertising routes.	**neighbor** { *neighbor-address* | *peer-group-name* } **next-hop-self**	Mandatory. By default, the next-hop property of the routes that are advertised to EBGP neighbors is set to the local IPv6 address, and the next-hop property of the routes that are advertised to IBGP neighbors keeps unchanged.

Note:

- When BGP is configured to use the local IPv6 address as the next hop of a route, if you run the **neighbor update-source** command to configure the source address of a TCP session, the source address is used as the next hop address; otherwise, the IP address of the output interface of the advertising device is selected as the local IPv6 address.

2. Configure a route map to modify NEXT-HOP properties.

BGP supports configuring a route map to modify NEXT-HOP properties. You can run the **set ipv6 next-hop** command to modify the next hop property.

Table 12-32 Configure a route map to modify NEXT-HOP properties

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-
Enter the BGP IPv6 unicast configuration mode	**address-family ipv6 unicast**	-

Step	Command	Description
Configure a route map to modify NEXT-HOP properties.	**neighbor** { *neighbor-address* | *peer-group-name* } **route-map** *rtmap-name* **in** | **out**	Mandatory. By default, no route map is applied to any neighbor.

Note:

- In configuring a route map to modify an NEXT-HOP property, you can use the **set ipv6 next-hop** command to modify the NEXT-HOP property. For details, refer to Routing Policy Tools-Command Manual-**set ipv6 next-hop**.

Configure the Community Property of a BGP Route

When BGP advertises routes to neighbors, it can be configured to send the community property. You can apply a route map to a specified neighbor in the incoming and outgoing directions to match the community properties.

Community property is used to identify a group of routes so as to apply a routing policy to the group of routes. Two types of community property are available: standard and extended. The standard community property consist of 4 bytes, providing the properties such as NO_EXPORT, LOCAL_AS, NO_ADVERTISE, and INTERNET. The extended property consist of eight bytes, providing Route Target (RT) and Route Origin (RO) properties.

1. Configure BGP to advertise route community property to neighbors.

The **neighbor send-community** enables you to advertise standard community property or extended community property or both types of property to neighbors.

Table 12-33 Configure BGP to advertise route community property to neighbors

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-
Enter the BGP IPv6 unicast configuration mode	**address-family ipv6 unicast**	-

Step	Command	Description
Configure BGP to advertise route community property to neighbors.	**neighbor** { *neighbor-address* | *peer-group-name* } **send-community** [**both** | **extended** | **standard**]	Mandatory. By default, the community property is not advertised to any neighbor.

Note:

- After neighbors are activated in VPNv6 address family, standard and extended community properties are automatically advertised to neighbors.

2. Configure a route map to modify the community property.

BGP supports configuring a route map to modify the route community property. You can use the **set communtiy** to command to modify the community property.

Table 12-34 Configure a route map to modify the community property

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-
Enter the BGP IPv6 unicast configuration mode	**address-family ipv6 unicast**	-
Configure a route map to modify the BGP route community property	**neighbor** { *neighbor-address* | *peer-group-name* } **route-map** *rtmap-name* **in** | **out**	Mandatory. By default, no route map is applied to any neighbor.

Note:

- In configuring a route map to modify community property, you can use the **set community** command to modify the community property. For details, refer to Routing Policy Tools-Command Manual-**set community**.

12.2.5. Configure IPv6 BGP Network Optimization

Configuration Condition

Before configuring BGP network optimization, ensure that:

- BGP is enabled.
- IPv6 BGP neighbors are configured and a session is set up successfully.

Configure the Keep-alive Time of BGP Neighbors

After a BGP session is successfully set up, keep-alive messages are sent periodically between the neighbors to maintain the BGP session. If no keep-alive message or Update packet is received from the neighbor within the hold time, the BGP session will be disconnected owing to timeout. The keep-alive time is equal to or smaller than 1/3 of the hold time.

Table 12-35 Configure the keep-alive time of BGP neighbors

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-
Configure global BGP keep-alive time and hold time.	**timers bgp** *keepalive-interval holdtime-interval*	Optional. By default, the keepalive timer is 60s, the hold timer is 180s, and the session re-connection timer is 120s.
Configure the keepalive time and hold time of a BGP neighbor or peer group.	**neighbor** { *neighbor-address* | *peer-group-name* } **timers** { *keepalive-interval holdtime-interval* | **connect** *connect-interval* }	

Note:

- The keepalive time and hold time that are set for a specified neighbor have higher priorities than the global BGP keepalive time and hold time.
- Neighbors negotiate and then take the minimum hold time as the hold of the BGP session between the neighbors.
- If the keepalive time and hold time are both set to 0, the neighbor keepalive/hold function is canceled.
- If the keepalive time is longer than 1/3 of the hold time, the BGP session sends keepalive packets at the interval of 1/3 the hold time.

Configure BGP Route Detection Time

BGP mainly aims at implementing a routing process, with ASs as the routing units. Within an AS, IGP is used for routing. Therefore, BGP routes often rely on IGP routes. If the next hops or output interfaces of IGP routes that BGP relies on change, BGP detects IGP routes periodically to update BGP routes. BGP also update local BGP routes during the detection interval.

Table 12-36 Configure BGP route detection time

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-
Configure BGP route detection time.	**bgp scan-time** *time*	Optional. By default, the BGP route detection time is 60s.

Caution:

- If the BGP route detection time is set too small, BGP detect routes frequently, affecting the device performance.

Configure Quick Disconnection of EBGP Neighbors

After a BGP session is successfully set up, Keepalive messages are sent periodically between the neighbors to maintain the BGP session. If no Keepalive message or Update packet is received from the neighbor within the hold time, the BGP session will be disconnected owing to timeout. You can configure direct-connect EBGP neighbors to disconnect a BGP connection immediately after a connecting interface is down, without waiting for BGP keepalive timeout. If the EBGP neighbor quick disconnection function is cancelled, the EBGP session does not respond to an interface down event; instead, the BGP session is disconnected after timeout.

Table 12-37 Configure quick disconnection of EBGP neighbors

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-
Configure quick disconnection of EBGP neighbors.	**bgp fast-external-failover**	Optional. By default, EBGP's quick processing capability in responding to the direct-connect interface down event is enabled.

Configure the BGP Route Suppression Function

Flapping routes in a network may cause instability of the network. You can configure route attenuation to damp this type of routes so as to decrease the effect of flapping routes on the network.

A frequently flapping route will be allocated with a penalty. If the penalty exceeds the suppression threshold, the route will not be advertised to neighbors. The penalty should not be kept beyond the maximum suppression time. If no flapping occurs on the route within the half-life period, the penalty will be halved. If the penalty is lower than the threshold value, the route can be advertised to neighbors again.

- Half-life period: It is the time in which the penalty of a route is halved.
- Reuse threshold: It is the threshold for the route to resume normal use.
- Suppression threshold: It is the threshold for route suppression.
- Maximum suppression time: It is the maximum threshold value for a route penalty.

Table 12-38 Configure the BGP route suppression function

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-
Enter the BGP IPv6 unicast configuration mode	**address-family ipv6 unicast**	-
Configure the BGP route attenuation period.	**bgp dampening** [*reach-half-life* [*reuse-value suppress-value max-suppress-time* [*unreach-half-life*]] | **route-map** *rtmap-name*]	Mandatory. By default, the route suppression function is disabled. After the function is enabled, the default route suppression half-life period is 15 minutes, the route reuse time is 750 seconds, the minimum route suppression time is 2000 seconds, the maximum route suppression time is 60 minutes, and the route penalty unreachable half-life period is 15 minutes.

Note:

- Route flapping not only contains addition and deletion of routes, but also contains route property changes such as next hop and MED property changes.

Configure the BGP Neighbor Refresh Capability

If the routing policy or route selection policy that is applied to a BGP neighbor changes, the route table needs to be refreshed. One way of refreshing the route table is to reset the BGP connection so as to reset the BGP session. However, this mode may result in BGP route flapping, affecting normal services. The other way is more graceful, that is, configuring the local BGP device to support the route refresh capability. If a neighbor needs to reset a route, it advertises the Route-Refresh message to the local device. After receiving the Route-Refresh message, it sends the route to the neighbor again. In this way, the route table is dynamically refreshed without the need of disconnecting the BGP session.

Table 12-39 Configure the BGP Neighbor refresh capability

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-
Enable the BGP neighbor refresh capability.	**neighbor** { *neighbor-address* | *peer-group-name* } **capability route-refresh**	Optional. By default, the BGP neighbor refresh capability is enabled.

Configure the BGP Neighbor Soft Reset Capability

If the routing policy or route selection policy that is applied to a BGP neighbor changes, the route table needs to be refreshed. One way of refreshing the route table is to reset the BGP connection so as to reset the BGP session. However, this mode may result in BGP route flapping, affecting normal services. Another way is more graceful, that is, configuring the local BGP device to support the route refresh capability. There is still another way, that is, enabling the soft reset capability of the local BGP device. By default, the BGP device reserves the routing information of each neighbor. After enabling its neighbor soft reset capability, it refreshes the neighbor routes that are kept on the local device. At this time, BGP sessions are not disconnected.

Table 12-40 Configure the BGP neighbor soft reset capability

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-
Enter the BGP IPv6 unicast configuration mode	**address-family ipv6 unicast**	-
Enable the BGP neighbor soft reset capability.	**neighbor** { *neighbor-address* | *peer-group-name* } **soft-reconfiguration inbound**	Mandatory. By default, the neighbor soft reset function is disabled.

Configure the ORF Capability of BGP Neighbors

BGP implements accurate route control through abundant routing properties. It usually applies routing policies in the incoming and outgoing directions. This mode is a local BGP behavior. BGP also supports the Outbound Route Filtering (ORF) capability. It advertises the local ingress policy to its neighbors through Route-refresh packets, and then the neighbors apply the policy when they advertise routes to the local BGP device. This greatly decreases the number of exchanged route refresh packets between BGP neighbors.

To achieve successful negotiation of the ORF capability, ensure that:

- The ORF capability is enabled for both neighbors.
- "ORF send" and "ORF receive" must match. That is, if one end is "ORF send", the other end must be "ORF both" or "ORF receive". If one end is "ORF receive", the other end must be "ORF send" or "ORF both".
- The "ORF send" end must be configured with a prefix list in the incoming direction.

Table 12-41 Configure the ORF capability of BGP neighbors

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-

Step	Command	Description
Enter the BGP IPv6 unicast configuration mode	**address-family ipv6 unicast**	-
Apply a prefix list in the incoming direction of a neighbor.	**neighbor** { *neighbor-address* | *peer-group-name* } **prefix-list** *prefix-list-name* **in**	Mandatory. By default, no prefix list is applied to any BGP neighbor.
Configure a neighbor to support the ORF capability.	**neighbor** { *neighbor-address* | *peer-group-name* } **capability orf prefix-list** { **both** | **receive** | **send** }	Mandatory. By default, a neighbor does not support the ORF capability.

12.2.6. Configure Large-Scale IPv6 BGP Network

Configuration Condition

Before configuring a large-scale BGP network, ensure that:

- BGP is enabled.
- IPv6 BGP neighbors are configured and a session is set up successfully.

Configure a BGP Peer Group

A BGP peer group is a group of BGP neighbors that are configured with the same configuration policy. Any configuration that is performed on a BGP peer group will take effect on all members of the peer group. In this way, by configuring the peer group, you can perform centralized management and maintenance on the neighbors.

Table 12-42 Configure a BGP peer group

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-
Enter the BGP IPv6 unicast configuration mode	**address-family ipv6 unicast**	-

Step	Command	Description
Create a BGP peer group.	**neighbor** *peer-group-name* **peer-group**	Mandatory. By default, no peer group is configured, and a neighbor is not in any peer group.
Add a neighbor into the peer group.	**neighbor** *neighbor-address* **peer-group** *peer-group-name*	

Note:

- The configuration on a peer group takes effect on all members of the peer group.
- After a neighbor is added into a peer group, if some configurations of the neighbor are the same as the configurations of the peer group, the configurations of the neighbor are deleted.
- If routing policies are configured in the incoming and outgoing directions of a peer group, after the routing policies are changed, the changes do not take effect on the neighbors that have been added into the peer group. To apply the changed routing policies on the peer group members, you need to reset the peer group.

Configure a BGP Route Reflector

In a large-scale BGP network, it is required that IBGP neighbors are fully connected, that is, each BGP needs to set up connections with all IBGP neighbors. In this way, in a network which contains N BGP neighbors, the number of BGP connections is N*(N-1)/2. The larger the number of connections is, the larger the number of route advertisements is. Configuring a BGP Route Reflector (RR) is a method of reducing the number of network connections. Multiple IBGPs are categorized into a group. In this group, a BGP is specified to act as the RR, while other BGPs act as client, and BGPs that are not in the group act as non-clients. Clients set up peer relations only with the RR while they do not set up peer relations with other BGPs. This reduces the number of mandatory IBGP connections, and the number of connections is N-1.

The following shoes the routing principles of the BGP RR:

- The RR reflects the routes that it learns from non-client IBGP neighbors only to clients.
- The RR reflects the routes that it learns from clients to all clients and non-clients except the clients that initiate the routes.
- The RR reflects the routes that it learns from EBGP neighbors to all clients and non-clients.

Table 12-43 Configure a BGP route reflector

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-

Step	Command	Description
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-
Configure an RR cluster ID.	**bgp cluster-id** { *cluster-id-in-ip* | *cluster-id-in-num* }	Mandatory. By default, the route ID is used as the RR cluster ID.
Configure a neighbor as a client of the RR.	**neighbor** { *neighbor-address* | *peer-group-name* } **route-reflector-client**	Mandatory. By default, no neighbor is specified as a client of the RR.
Configure the route reflection function between BGP clients.	**bgp client-to-client reflection**	Optional. By default, the route reflection function is enabled between RR clients.

Note:

- An RR cluster ID is used to identify an RR area. An RR area can contain multiple RRs, and the RRs in the RR area have the same RR cluster ID.

Configure a BGP Confederation

In a large-scale BGP network, it is required that IBGP neighbors are fully connected, that is, each BGP needs to set up connections with all IBGP neighbors. In this way, in a network which contains N BGP neighbors, the number of BGP connections is N*(N-1)/2. The larger the number of connections is, the larger the number of route advertisements is. Configuring BGP confederations is another way of reducing the number of network connections. An AS area is divided into multiple sub-AS areas, and each AS area forms a confederation. IBGP is adopted within a confederation to provide full connections, and sub-AS areas in the confederation are connected through EBGP connections. This effectively reduces the number of BGP connections.

In configuring BGP confederations, you need to assign a confederation ID for each confederation and specify members for the confederation. In the case of route reflection, only the route reflector is required to support route reflection. However, in the case of a confederation, all members in a confederation must support the confederation function.

Table 12-44 Configure a BGP confederation

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-
Create a BGP confederation ID.	**bgp confederation identifier** *as-number*	Mandatory. By default, no AS number is configured for a confederation.
Configure members for the confederation.	**bgp confederation peers** *as-number-list*	Mandatory. By default, no sub-AS number is configured for a confederation.

Note:

- A confederation ID is used to identify the sub-ASs of the confederation. Confederation members are divided into the sub-ASs.

12.2.7. Configure IPv6 BGP to Coordinate with BFD

Usually, there are still some intermediate devices between BGP neighbors. When an intermediate device becomes faulty, the BGP session is normal within the hold time, and the link fault caused by the intermediate device cannot be responded to in time. Bidirectional Forwarding Detection (BFD) provides a method for quickly detecting the status of a line between two devices. After BFD is enabled for BGP devices, if a line between two devices becomes faulty, BFD can quickly find the line fault and notifies BGP of the fault. It triggers BGP to quickly disconnect the session and quickly switch over to the backup line, achieving fast switchover of routes.

Configuration Condition

Before configuring BGP to coordinate with BFD, ensure that:

- BGP is enabled.
- IPv6 BGP neighbors are configured and a session is set up successfully.

Configure EBGP to Coordinate with BFD

The coordination between EBGP and BFD is based on a single-hop BFD session, and BFD session parameters need to be configured in interface mode.

Table 12-45 Configure EBGP to coordinate with BFD

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-
Enter the BGP IPv6 unicast configuration mode	**address-family ipv6 unicast**	-
Configure EBGP to coordinate with BFD.	**neighbor** { *neighbor-address* | *peer-group-name* } **fall-over bfd** [**single-hop**]	Mandatory. By default, the BFD function is disabled for a neighbor.
Exit the BGP IPv6 unicast configuration mode	**exit-address-family**	-
Exit the BGP configuration mode.	**exit**	-
Enter the interface configuration mode.	**interface** *interface-name*	-
Configure the minimum receive interval of a BFD session.	**bfd min-receive-interval** *milliseconds*	Optional. By default, the minimum receive interval of a BFD session is 1000ms.
Configure the minimum transmit interval of the BFD session.	**bfd min-transmit-interval** *milliseconds*	Optional. By default, the minimum transmit interval of a BFD session is 1000ms.

Step	Command	Description
Configure the multiple of BFD session detection timeout.	**bfd multiplier** *number*	Optional. By default, the multiple of BFD session detection timeout is 5.

Note:

- For the related configuration of BFD, refer to the reliability technology-BFD technical manual and BFD configuration manual.

Configure IBGP to Coordinate with BFD

The coordination between IBGP and BFD is based on a multi-hop BFD session, and BFD session parameters need to be configured in BGP mode.

Table 12-46 Configure IBGP to coordinate with BFD

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-
Enter the BGP IPv6 unicast configuration mode	**address-family ipv6 unicast**	-
Configure IBGP to coordinate with BFD.	**neighbor** { *neighbor-address* | *peer-group-name* } **fall-over bfd** [**single-hop**]	Mandatory. By default, the BFD function is disabled for a neighbor.
Configure the minimum receive interval of the BFD session.	**bfd min-receive-interval** *milliseconds*	Optional. By default, the minimum receive interval of a BFD session is 1000ms.
Configure the minimum transmit interval of the BFD session.	**bfd min-transmit-interval** *milliseconds*	Optional. By default, the minimum transmit interval of a BFD session is 1000ms.

Step	Command	Description
Configure the multiple of BFD session detection timeout.	**bfd multiplier** *number*	Optional. By default, the multiple of BFD session detection timeout is 5.

12.2.8. Configure IPv6 BGP Fast Re-routing

Configuration Conditions

Before configuring IPv6 BGP fast re-routing, ensure that:

- When configuring fast rerouting based on route-map, the associated route-map has been configured.
- Enable the IPv6BG protocol.

Configure BGP Fast Re-routing

In the IPv6 BGP network, if the link or device fails, the packet passing the fault point will be dropped or generate the loop and the caused traffic interruption will not recover until the protocol re-converges, which often lasts for several seconds. To reduce the traffic interruption time, you can configure the IPv6 BGP fast re-routing. Apply the route map to set the backup next hop for the matched route. Once the active link fails, the traffic passing the faulty link will switch to the standby link at once, so as to realize fast switching.

Table 12-47 Configure the IPv6 BGP fast re-routing

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-
Enter the BGP IPv6 unicast configuration mode	**address-family ipv6 unicast**	-
Configure fast re-routing based on route-map	**fast-reroute route-map** *route-map-name*	Mandatory By default, do not enable the fast re-routing function based on route-map.

Step	Command	Description
Configure auto fast re-routing of BGP	**pic**	Mandatory By default, do not enable the auto fast re-routing function.

Caution:

- After configuring the BGP fast re-routing, you need to re-set BGP and complete the checking and backup of the initial route. Otherwise, it takes effect only for the route learned after configuration.
- For fast re-routing based on route-map, when configuring set fast-reroute backup-nexthop auto, the protocol performs auto fast re-routing.
- When using the pic mode, the protocol performs the auto fast re-routing.
- The various modes of enabling the fast re-routing are mutually exclusive.
- After configuring the BGP fast re-routing to apply the route map, set the BGP neighbor as the backup next hop via the **set fast-reroute backup-nexthop** *nexthop-address* command. If configuring the non-BGP neighbor as the backup next hop, you cannot make the fast re-routing function take effect.

12.2.9. Configure BGP Neighbor Protection

Configuration Conditions

Before configuring the BGP neighbor protection, first complete the following task:

- Create ipsec tunnel used to protect the neighbor
- Enable the BGP protocol
- Configure the BGP protocol and set up the session connection successfully.

Configure BGP Neighbor Protection

In BGP network, because it is plaintext communication, it is easy to receive monitoring and attacks from the network. BGP neighbor protection is to solve such problems. IPSec tunnel is used to protect the communication data between neighbors to avoid data being monitored and attacked.

Table 12-48 Configure the BGP neighbor protection

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the BGP configuration mode.	**router bgp** *autonomous-system*	-

Step	Command	Description
Enter the BGP IPv6 unicast configuration mode	**address-family ipv6 unicast**	-
Configure the BGP neighbor protection	**neighbor** { *neighbor-address* | *peer-group-name* } **ipsec-tunnel** *tunnel-name*	Optional By default, do not enable the neighbor protection function.

12.2.10. IPv6 BGP Monitoring and Maintaining

Table 12-49 IPv6 BGP monitoring and maintaining

Command	Description
clear bgp ipv6 { * | *as-number* | **peer-group** *peer-group-name* | **external** | *neighbor-address* } [**vrf** *vrf-name*]	Resets the BGP neighbor.
clear bgp [**ipv6 unicast**] **dampening** [*ipv6-address* | *ipv6-address/mask-length*]	Clears suppressed routes.
clear bgp [**ipv6 unicast**] **flap-statistics** [*ipv6-address* | *ipv6-address/mask-length*]	Clears the flap statistics information
clear bgp [**ipv6**] { * | *as-number* | **peer-group** *peer-group-name* | **external** | *neighbor-address*} [**vrf** *vrf-name*] { [**soft**] [**in** | **out**] }	Soft-resets neighbors.
clear bgp [**ipv6**] { * | *neighbor-address* | *as-number* | **peer-group** *peer-group-name* | **external** } [**vrf** *vrf-name*] **in prefix-filter**	Advertises ORF to neighbors.
show bgp{ipv6 unicast | vpnv6 unicast { **all** | **vrf** *vrf-name* | **rd** *route-distinguisher* } } [*ipv6-address* | *ipv6-address/mask-length*]	Displays the routing information in the related BGP address family.

Command	Description
show ip bgp attribute-info	Displays the BGP common route attributes.
show bgp ipv6 unicast community [*community-number* / *aa:nn* / **exact-match** / **local-AS** / **no-advertise** / **no-export**]	Displays the routes that match the specified community property.
show bgp ipv6 unicast community-list *community-list-name*	Displays the community list that is applied to routes.
show bgp {**ipv6 unicast | vpnv6 unicast** { **all** | **vrf** *vrf-name* | **rd** *route-distinguisher* } } **dampening** { **dampened-paths | flap-statistics | parameters** }	Displays the details of route attenuation.
show bgp ipv6 unicast filter-list *filter-list-name* [**exact-match**]	Displays the routes that match the AS_PATH filter list.
show bgp ipv6 unicast inconsistent-as	Displays the routes that conflict with AS_PATH.
show bgp { **ipv6 unicast | vpnv6 uicast** **{ all** | **vrf** *vrf-name* | **rd** *route-distinguisher* **}} neighbors** [*ipv6-address*]	Displays the details of the BGP neighbors
show bgp ipv6 unicast prefix-list *prefix-list-name*	Displays the routes that match the prefix list
show bgp ipv6 unicast quote-regexp *as-path-list-name*	Displays the routes that match the AS_PATH list
show bgp ipv6 unicast regexp *as-path-list-name*	Displays the routes that match the AS_PATH list
show bgp ipv6 unicast route-map *rtmap-name*	Displays the routes that match the route map

Command	Description
show ip bgp scan	Displays the BGP scan information.
show bgp {**ipv6 unicast** | **vpnv6** { **all** | **vrf** *vrf-name* | **rd** *route-distinguisher* } } **summary**	Displays the summary of BGP neighbors.

12.3. IPv6 BGP Typical Configuration Example

12.3.1. Configure IPv6 BGP Basic Functions

Network Requirements

- Set up EBGP neighbors between Device1 and Device2, and set up IBGP neighbors between Device2 and Device3.
- Device1 learns the interface direct route 2001:4::/64 of Device3, and Device3 learns the interface direct route 2001:1::/64 of Device1.

Network Topology

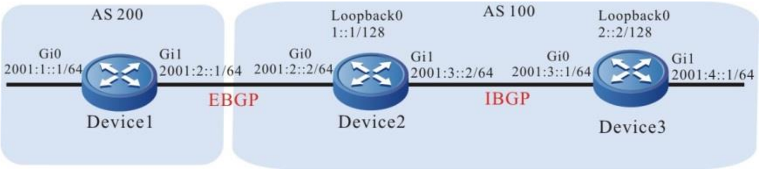

Figure 12–1 Networking for configuring IPv6 BGP basic functions

Configuration Steps

Step 1: Configure the IPv6 global unicast addresses of the interfaces. (Omitted)

Step 2: Configure OSPFv3 so that loopback routes are reachable between devices.

#Configure Device2.

```
Device2#configure terminal
Device2(config)#ipv6 router ospf 100
Device2(config-ospf6)#router-id 2.2.2.2
Device2(config-ospf6)#exit
Device2(config)#interface gigabitethernet 1
Device2(config-if-gigabitethernet1)#ipv6 router ospf 100 area 0
Device2(config-if-gigabitethernet1)#exit
Device2(config)#interface loopback 0
Device2(config-if-loopback0)#ipv6 router ospf 100 area 0
Device2(config-if-loopback0)#exit
```

#Configure Device3.

```
Device3#configure terminal
```

```
Device3(config)#ipv6 router ospf 100
Device3(config-ospf6)#router-id 3.3.3.3
Device3(config-ospf6)#exit
Device3(config)#interface gigabitethernet 0
Device3(config-if-gigabitethernet0)#ipv6 router ospf 100 area 0
Device3(config-if-gigabitethernet0)#exit
Device3(config)#interface loopback 0
Device3(config-if-loopback0)#ipv6 router ospf 100 area 0
Device3(config-if-loopback0)#exit
```

#View the route table of Device2.

```
Device2#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management

L   ::1/128 [0/0]
    via ::, 1w1d:23:51:37, lo0
LC  1::1/128 [0/0]
    via ::, 00:09:34, loopback0
O   2::2/128 [110/2]
    via fe80::201:7aff:fec0:525a, 00:05:29, gigabitethernet1
C   2001:2::/64 [0/0]
    via ::, 00:09:41, gigabitethernet0
L   2001:2::2/128 [0/0]
    via ::, 00:09:39, lo0
C   2001:3::/64 [0/0]
    via ::, 00:08:55, gigabitethernet1
L   2001:3::2/128 [0/0]
    via ::, 00:08:53, lo0
```

#View the route table of Device3.

```
Device3#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management
```

```
L   ::1/128 [0/0]
    via ::, 1w5d:18:34:53, lo0
O   1::1/128 [110/2]
    via fe80::201:7aff:fe5e:6d2e, 00:29:59, gigabitethernet0
LC  2::2/128 [0/0]
    via ::, 00:32:36, loopback0
C   2001:3::/64 [0/0]
    via ::, 00:32:59, gigabitethernet0
L   2001:3::1/128 [0/0]
    via ::, 00:32:58, lo0
C   2001:4::/64 [0/0]
    via ::, 00:32:44, gigabitethernet1
L   2001:4::1/128 [0/0]
    via ::, 00:32:43, lo0
```

According to the queried information, Device2 and Device3 have learnt the routes of the peer loopback interfaces by running OSPFv3, preparing for setting up IBGP neighbors on the loopback interfaces of Device2 and Device3.

Step 3: Configure the IPv6 BGP basic functions.

#Configure Device1.

Set up a direct-connect EBGP peer with Device2. Introduce 2001:1::/64 to BGP in network mode.

```
Device1#configure terminal
Device1(config)#router bgp 200
Device1(config-bgp)#bgp router-id 1.1.1.1
Device1(config-bgp)#address-family ipv6
Device1(config-bgp-af)#neighbor 2001:2::2 remote-as 100
Device1(config-bgp-af)#network 2001:1::/64
Device1(config-bgp-af)#exit-address-family
Device1(config-bgp)#exit
```

#Configure Device2.

Set up the direct-connect EBGP peer with Device1, set up a non-direct-connect IBGP peer with Device3 through Loopback0, and set the next hop of the advertised route to the local device.

```
Device2(config)#router bgp 100
Device2(config-bgp)#bgp router-id 2.2.2.2
Device2(config-bgp)#address-family ipv6
Device2(config-bgp-af)#neighbor 2001:2::1 remote-as 200
Device2(config-bgp-af)#neighbor 2::2 remote-as 100
Device2(config-bgp-af)#neighbor 2::2 next-hop-self
```

```
Device2(config-bgp-af)#exit-address-family
Device2(config-bgp)#neighbor 2::2 update-source loopback 0
Device2(config-bgp)#exit
```

#Configure Device3.

Set up a non-direct-connect IBGP peer relation with Device2 through Loopback0. Introduce 2001:4::/64 to BGP in network mode.

```
Device3(config)#router bgp 100
Device3(config-bgp)#bgp router-id 3.3.3.3
Device3(config-bgp)#address-family ipv6
Device3(config-bgp-af)#neighbor 1::1 remote-as 100
Device3(config-bgp-af)#network 2001:4::/64
Device3(config-bgp-af)#exit-address-family
Device3(config-bgp)#neighbor 1::1 update-source loopback 0
Device3(config-bgp)#exit
```

Note:

- To prevent route flapping, IBGP neighbors are set up through the loopback interfaces, and OSPFv3 need to synchronize the routing information of loopback interfaces between IBGP neighbors.

Step 4: Check the result.

#On Device2, check the IPv6 BGP neighbor status.

```
Device2#show bgp ipv6 unicast summary
BGP router identifier 2.2.2.2, local AS number 100
BGP table version is 4
2 BGP AS-PATH entries
0 BGP community entries

Neighbor      V   AS MsgRcvd MsgSent  TblVer  InQ OutQ Up/Down  State/PfxRcd
2::2         4  100     8      6       3    0    0 00:04:12       1
2001:2::1     4  200     15     15      3    0    0 00:11:17       1

Total number of neighbors 2
```

According to the numbers (Number of route prefixes received from neighbors) that are displayed in the State/PfxRcd column, IPv6 BGP neighbors have been successfully set up between Device 2 and Device 1, Device 3.

#View the route table of Device1.

```
Device1#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
```

```
    O - OSPF, OE-OSPF External, M - Management

L  ::1/128 [0/0]
   via ::, 1w2d:00:42:57, lo0
C  2001:1::/64 [0/0]
   via ::, 00:02:59, gigabitethernet0
L  2001:1::1/128 [0/0]
   via ::, 00:02:56, lo0
C  2001:2::/64 [0/0]
   via ::, 00:52:17, gigabitethernet1
L  2001:2::1/128 [0/0]
   via ::, 00:52:16, lo0
B  2001:4::/64 [20/0]
   via 2001:2::2, 00:06:13, gigabitethernet1
```

#View the route table of Device2.

```
Device2#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
    U - Per-user Static route
    O - OSPF, OE-OSPF External, M - Management

L  ::1/128 [0/0]
   via ::, 1w2d:00:34:53, lo0
LC  1::1/128 [0/0]
   via ::, 00:52:49, loopback0
O  2::2/128 [110/2]
   via fe80::201:7aff:fec0:525a, 00:48:45, gigabitethernet1
B  2001:1::/64 [20/0]
   via 2001:2::1, 00:03:18, gigabitethernet0
C  2001:2::/64 [0/0]
   via ::, 00:52:57, gigabitethernet0
L  2001:2::2/128 [0/0]
   via ::, 00:52:55, lo0
C  2001:3::/64 [0/0]
   via ::, 00:52:10, gigabitethernet1
L  2001:3::2/128 [0/0]
   via ::, 00:52:09, lo0
B  2001:4::/64 [200/0]
```

```
    via 2::2, 00:07:27, gigabitethernet1
```

#View the route table of Device3.

```
Device3#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
      U - Per-user Static route
      O - OSPF, OE-OSPF External, M - Management

L   ::1/128 [0/0]
    via ::, 1w5d:18:54:38, lo0
O   1::1/128 [110/2]
    via fe80::201:7aff:fe5e:6d2e, 00:49:44, gigabitethernet0
LC  2::2/128 [0/0]
    via ::, 00:52:21, loopback0
B   2001:1::/64 [200/0]
    via 1::1, 00:03:54, gigabitethernet0
C   2001:3::/64 [0/0]
    via ::, 00:52:44, gigabitethernet0
L   2001:3::1/128 [0/0]
    via ::, 00:52:43, lo0
C   2001:4::/64 [0/0]
    via ::, 00:52:29, gigabitethernet1
L   2001:4::1/128 [0/0]
    via ::, 00:52:28, lo0
```

Device1 has learnt the interface direct-connect route 2001:4::/64 of Device3, and Device3 has learnt the interface direct-connect route 2001:1::/64 of Device1.

12.3.2. Configure IPv6 BGP to Re-distribute Routes

Network Requirements

- Set up OSPFv3 neighbors between Device3 and Device2, and advertise interface direct-connect route 2001:3::/64 to Device2.
- Set up EBGP neighbors between Device1 and Device2, and redistribute the OSPFv3 route that Device2 learns to BGP and advertise the route to Device1.

Network Topology

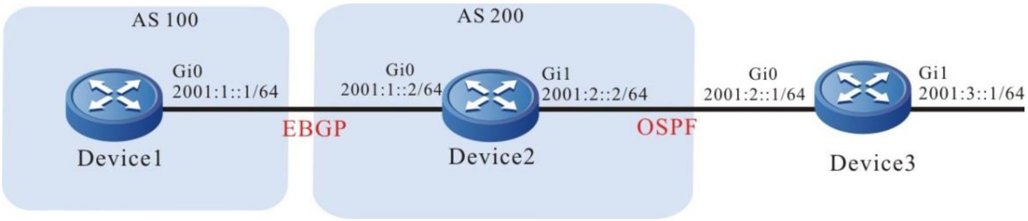

Figure 12–2 Networking for configuring IPv6 BGP to redistribute routes

Configuration Steps

Step 1: Configure the IPv6 global unicast addresses of the interfaces. (Omitted)

Step 2: Configure OSPFv3 so that Device2 can learn the direct-connect interface route 2001:3::/64 to Device3.

#Configure Device2.

```
Device2#configure terminal
Device2(config)#ipv6 router ospf 100
Device2(config-ospf6)#router-id 2.2.2.2
Device2(config-ospf6)#exit
Device2(config)#interface gigabitethernet 1
Device2(config-if-gigabitethernet1)#ipv6 router ospf 100 area 0
Device2(config-if-gigabitethernet1)#exit
```

#Configure Device3.

```
Device3#configure terminal
Device3(config)#ipv6 router ospf 100
Device3(config-ospf6)#router-id 3.3.3.3
Device3(config-ospf6)#exit
Device3(config)#interface gigabitethernet 0
Device3(config-if-gigabitethernet0)#ipv6 router ospf 100 area 0
Device3(config-if-gigabitethernet0)#exit
Device3(config)#interface gigabitethernet 1
Device3(config-if-gigabitethernet1)#ipv6 router ospf 100 area 0
Device3(config-if-gigabitethernet1)#exit
```

#View the route table of Device2.

```
Device2#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management

L   ::1/128 [0/0]
    via ::, 1w2d:01:10:38, lo0
C   2001:1::/64 [0/0]
    via ::, 00:06:25, gigabitethernet0
L   2001:1::2/128 [0/0]
    via ::, 00:06:24, lo0
C   2001:2::/64 [0/0]
    via ::, 00:05:46, gigabitethernet1
```

```
L   2001:2::2/128 [0/0]
    via ::, 00:05:43, lo0
O   2001:3::/64 [110/2]
    via fe80::201:7aff:fec0:525a, 00:02:41, gigabitethernet1
```

According to the route table, Device2 has learnt the OSPFv3 route 2001:3::/64 that has been advertised by Device3.

Step 3: Configure the IPv6 BGP basic functions.

#Configure Device1.

```
Device1#configure terminal
Device1(config)#router bgp 100
Device1(config-bgp)#bgp router-id 1.1.1.1
Device1(config-bgp)#address-family ipv6
Device1(config-bgp-af)#neighbor 2001:1::2 remote-as 200
Device1(config-bgp-af)#exit-address-family
Device1(config-bgp)#exit
```

#Configure Device2.

```
Device2(config)#router bgp 200
Device2(config-bgp)#bgp router-id 2.2.2.2
Device2(config-bgp)#address-family ipv6
Device2(config-bgp-af)#neighbor 2001:1::1 remote-as 100
Device2(config-bgp-af)#exit-address-family
Device2(config-bgp)#exit
```

#On Device2, check the IPv6 BGP neighbor status.

```
Device2#show bgp ipv6 unicast summary
BGP router identifier 2.2.2.2, local AS number 200
BGP table version is 1
1 BGP AS-PATH entries
0 BGP community entries

Neighbor      V    AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down  State/PfxRcd
2001:1::1     4   100       2       2        1    0    0 00:00:50        0

Total number of neighbors 1
```

IPv6 BGP neighbors have been successfully set up between Device2 and Device1.

Step 4: Configure IPv6 BGP to redistribute the OSPFv3 route.

#Configure Device2.

```
Device2(config)#router bgp 200
Device2(config-bgp)#address-family ipv6
Device2(config-bgp-af)#redistribute ospf 100
Device2(config-bgp-af)#exit-address-family
Device2(config-bgp)#exit
```

Step 5: Check the result.

#View the IPv6 BGP route table of Device2.

```
Device2#show bgp ipv6 unicast
BGP table version is 2, local router ID is 2.2.2.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
          S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
    Network          Next Hop          Metric    LocPrf Weight Path
[O]*> 2001:2::/64      ::                1          32768 ?
[O]*> 2001:3::/64      ::                2          32768 ?
```

According to the queried information, OSPFv3 routes have been successfully redistributed to IPv6 BGP.

#View the route table of Device1.

```
Device1#show bgp ipv6 unicast
BGP table version is 3, local router ID is 1.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
          S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
    Network          Next Hop          Metric    LocPrf Weight Path
[B]*> 2001:2::/64      2001:1::2           1              0 200 ?
[B]*> 2001:3::/64      2001:1::2           2              0 200 ?
```

According to the queried information, Device1 has successfully learnt routes 2001:2::/64 and 2001:3::/64.

Note:

- In an actual application, if there are two or more AS boundary routers, it is recommended that you do not redistribute routes between different routing protocols. If route redistribution must be configured, you are required to configure route control policies such as route filtering and filtration summary on the AS boundary routers to prevent routing loops.

12.3.3. Configure IPv6 BGP Community Properties

Network Requirements

- Set up EBGP neighbors between Device1 and Device2.
- Device1 introduces two direct-connect routes 2001:1::/64 and 2001:2::/64 to BGP in network mode, and set different community properties for two routes that are advertised to Device2.
- When Device2 receives routes from Device1, it applies community properties in the incoming direction of a neighbor to filter route 2001:1::/64 and allow route 2001:2::/64.

Network Topology

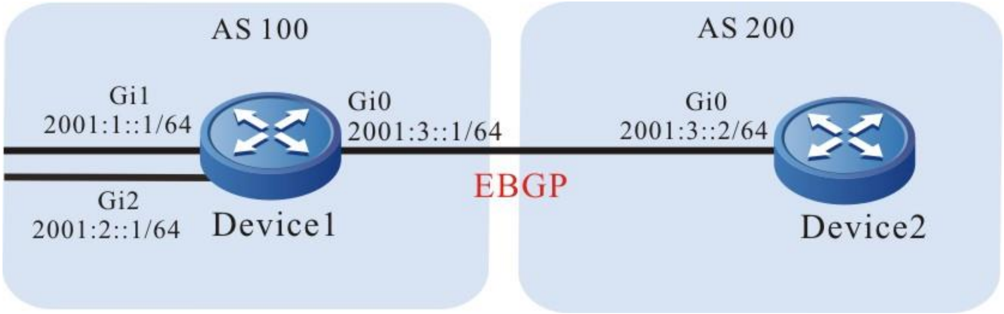

Figure 12–3 Networking for configuring IPv6 BGP community properties

Configuration Steps

Step 1: Configure the IPv6 global unicast addresses of the interfaces. (Omitted)

Step 2: Configure the IPv6 BGP basic functions.

#Configure Device1.

```
Device1#configure terminal
Device1(config)#router bgp 100
Device1(config-bgp)#bgp router-id 1.1.1.1
Device1(config-bgp)#address-family ipv6
Device1(config-bgp-af)#neighbor 2001:3::2 remote-as 200
Device1(config-bgp-af)#network 2001:1::/64
Device1(config-bgp-af)#network 2001:2::/64
Device1(config-bgp-af)#exit-address-family
Device1(config-bgp)#exit
```

#Configure Device2.

```
Device2#configure terminal
Device2(config)#router bgp 200
```

```
Device2(config-bgp)#bgp router-id 2.2.2.2
Device2(config-bgp)#address-family ipv6
Device2(config-bgp-af)#neighbor 2001:3::1 remote-as 100
Device2(config-bgp-af)#exit-address-family
Device2(config-bgp)#exit
```

#On Device1, check the IPv6 BGP neighbor status.

```
Device1#show bgp ipv6 unicast summary
BGP router identifier 1.1.1.1, local AS number 100
BGP table version is 1
1 BGP AS-PATH entries
0 BGP community entries

Neighbor       V    AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down  State/PfxRcd
2001:3::2      4   200       3       4        1    0    0 00:01:02        0

Total number of neighbors 1
```

IPv6 BGP neighbors have been successfully set up between Device1 and Device2.

#Query the route table of Device2.

```
Device2#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management

L   ::1/128 [0/0]
    via ::, 1w2d:05:45:34, lo0
B   2001:1::/64 [20/0]
    via 2001:3::1, 00:01:35, gigabitethernet0
B   2001:2::/64 [20/0]
    via 2001:3::1, 00:01:35, gigabitethernet0
C   2001:3::/64 [0/0]
    via ::, 00:04:09, gigabitethernet0
L   2001:3::2/128 [0/0]
    via ::, 00:04:08, lo0
```

According to the queried information, Device2 has successfully learnt routes 2001:1::/64 and 2001:2::/64.

Step 3: Configure the ACL and routing policy, and set IPv6 BGP community properties.

#Configure Device1.

```
Device1(config)#ipv6 access-list extended 7001
Device1(config-v6-list)#permit ipv6 2001:1::/64 any
Device1(config-v6-list)#exit
Device1(config)#ipv6 access-list extended 7002
Device1(config-v6-list)#permit ipv6 2001:2::/64 any
Device1(config-v6-list)#exit
Device1(config)#route-map CommunitySet 10
Device1(config-route-map)#match ipv6 address 7001
Device1(config-route-map)#set community 100:1
Device1(config-route-map)#exit
Device1(config)#route-map CommunitySet 20
Device1(config-route-map)#match ipv6 address 7002
Device1(config-route-map)#set community 100:2
Device1(config-route-map)#exit
```

Set different community properties for routes 2001:1::/64 and 2001:2::/64 respectively by configuring an ACL and routing policy.

Step 4: Configure a routing policy for IPv6 BGP.

#Configure Device1.

```
Device1(config)#router bgp 100
Device1(config-bgp)#address-family ipv6
Device1(config-bgp-af)#neighbor 2001:3::2 route-map CommunitySet out
Device1(config-bgp-af)#neighbor 2001:3::2 send-community
Device1(config-bgp-af)#exit-address-family
Device1(config-bgp)#exit
```

#View the IPv6 BGP route table of Device2.

```
Device2#show bgp ipv6 unicast 2001:1::/64
BGP routing table entry for 2001:1::/64
Paths: (1 available, best #1, table Default-IP-Routing-Table)
  Not advertised to any peer
  100
    2001:3::1 (metric 10) from 2001:3::1 (1.1.1.1)



     Origin IGP, metric 0, localpref 100, valid, external, best
      Community: 100:1
      Last update: 00:00:24 ago
Device2#show bgp ipv6 unicast 2001:2::/64
```

```
BGP routing table entry for 2001:2::/64
Paths: (1 available, best #1, table Default-IP-Routing-Table)
 Not advertised to any peer
 100
  2001:3::1 (metric 10) from 2001:3::1 (1.1.1.1)



   Origin IGP, metric 0, localpref 100, valid, external, best
    Community: 100:2
    Last update: 00:00:30 ago
```

According to the IPv6 BGP route table of Device2, the community property of route 2001:1::/64 is set to 100:1, and the community properties of route 2001:2::/64 is set to 100:2.

Step 5: Configure IPv6 BGP route filtration.

#Configure Device2.

```
Device2(config)#ip community-list 1 permit 100:2
Device2(config)#route-map CommunityFilter
Device2(config-route-map)#match community 1
Device2(config-route-map)#exit
Device2(config)#router bgp 200
Device2(config-bgp)#address-family ipv6
Device2(config-bgp-af)#neighbor 2001:3::1 route-map CommunityFilter in
Device2(config-bgp-af)#exit-address-family
Device2(config-bgp)#exit
```

Step 6: Check the result.

#View the route table of Device2.

```
Device2#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management


L   ::1/128 [0/0]
    via ::, 1w2d:05:58:57, lo0
B   2001:2::/64 [20/0]
    via 2001:3::1, 00:00:05, gigabitethernet0
C   2001:3::/64 [0/0]
    via ::, 00:17:32, gigabitethernet0
```

```
L   2001:3::2/128 [0/0]
    via ::, 00:17:30, lo0
```

According to the IPv6 BGP route table of Device2, route 2001:1::/64 has been filtered in the incoming direction, and route 2001:2::/64 has been allowed.

Note:

- After a routing policy is configured on the IPv6 BGP neighbor, the IPv6 BGP must be reset to make the configuration take effect.
- You must configure the **send-community** command to advertise the community property to the peer.

12.3.4. Configure IPv6 BGP Route Reflector

Network Requirements

- Set up EBGP neighbors between Device3 and Device4, and configure Device4 to advertise route 2001:4::/64.
- Set up IBGP neighbors between Device2 and Device3 and between Device2 and Device1 respectively. On Device2, configure Route Reflectors (RRs), and configure Device1 and Device3 as clients, so that Device1 can learn route 2001:4::/64 that is advertised by Device4.

Network Topology

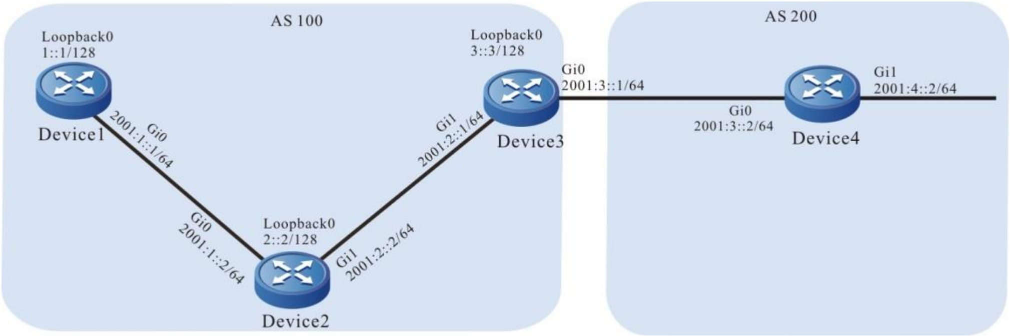

Figure 12–4 Networking for configuring an IPv6 BGP route reflector

Configuration Steps

Step 1: Configure the IPv6 global unicast addresses of the interfaces. (Omitted)

Step 2: Configure OSPFv3 so that loopback routes are reachable between devices.

#Configure Device1.

```
Device1#configure terminal
Device1(config)#ipv6 router ospf 100
Device1(config-ospf6)#router-id 1.1.1.1
Device1(config-ospf6)#exit
Device1(config)#interface gigabitethernet 0
Device1(config-if-gigabitethernet0)#ipv6 router ospf 100 area 0
Device1(config-if-gigabitethernet0)#exit
Device1(config)#interface loopback 0
```

```
Device1(config-if-loopback0)#ipv6 router ospf 100 area 0
Device1(config-if-loopback0)#exit
```

#Configure Device2.

```
Device2#configure terminal
Device2(config)#ipv6 router ospf 100
Device2(config-ospf6)#router-id 2.2.2.2
Device2(config-ospf6)#exit
Device2(config)#interface gigabitethernet 0
Device2(config-if-gigabitethernet0)#ipv6 router ospf 100 area 0
Device2(config-if-gigabitethernet0)#exit
Device2(config)#interface gigabitethernet 1
Device2(config-if-gigabitethernet1)#ipv6 router ospf 100 area 0
Device2(config-if-gigabitethernet1)#exit
Device2(config)#interface loopback 0
Device2(config-if-loopback0)#ipv6 router ospf 100 area 0
Device2(config-if-loopback0)#exit
```

#Configure Device3.

```
Device3#configure terminal
Device3(config)#ipv6 router ospf 100
Device3(config-ospf6)#router-id 3.3.3.3
Device3(config-ospf6)#exit
Device3(config)#interface gigabitethernet 1
Device3(config-if-gigabitethernet1)#ipv6 router ospf 100 area 0
Device3(config-if-gigabitethernet1)#exit
Device3(config)#interface loopback 0
Device3(config-if-loopback0)#ipv6 router ospf 100 area 0
Device3(config-if-loopback0)#exit
```

#View the route table of Device1.

```
Device1#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management

L   ::1/128 [0/0]
    via ::, 1w2d:06:26:16, lo0
LC  1::1/128 [0/0]
    via ::, 00:13:56, loopback0
```

```
O   2::2/128 [110/2]
    via fe80::201:7aff:fec0:525a, 00:09:06, gigabitethernet0
O   3::3/128 [110/3]
    via fe80::201:7aff:fec0:525a, 00:00:36, gigabitethernet0
C   2001:1::/64 [0/0]
    via ::, 00:14:03, gigabitethernet0
L   2001:1::1/128 [0/0]
    via ::, 00:14:02, lo0
O   2001:2::/64 [110/2]
    via fe80::201:7aff:fec0:525a, 00:09:06, gigabitethernet0
```

#View the route table of Device2.

```
Device2#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
     U - Per-user Static route
     O - OSPF, OE-OSPF External, M - Management

L   ::1/128 [0/0]
    via ::, 1w6d:00:46:09, lo0
O   1::1/128 [110/2]
    via fe80::201:7aff:fe5e:6d2e, 00:10:05, gigabitethernet0
LC  2::2/128 [0/0]
    via ::, 00:14:23, loopback0
O   3::3/128 [110/2]
    via fe80::201:7aff:fe62:bb80, 00:01:44, gigabitethernet1
C   2001:1::/64 [0/0]
    via ::, 00:14:48, gigabitethernet0
L   2001:1::2/128 [0/0]
    via ::, 00:14:47, lo0
C   2001:2::/64 [0/0]
    via ::, 00:14:41, gigabitethernet1
L   2001:2::2/128 [0/0]
    via ::, 00:14:39, lo0
```

#View the route table of Device3.

```
Device3#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
     U - Per-user Static route
     O - OSPF, OE-OSPF External, M - Management
```

```
L   ::1/128 [0/0]
    via ::, 1w2d:06:37:24, lo0
O   1::1/128 [110/3]
    via fe80::201:7aff:fec0:525b, 00:02:39, gigabitethernet1
O   2::2/128 [110/2]
    via fe80::201:7aff:fec0:525b, 00:02:39, gigabitethernet1
LC  3::3/128 [0/0]
    via ::, 00:14:45, loopback0
O   2001:1::/64 [110/2]
    via fe80::201:7aff:fec0:525b, 00:02:39, gigabitethernet1
C   2001:2::/64 [0/0]
    via ::, 00:15:03, gigabitethernet1
L   2001:2::1/128 [0/0]
    via ::, 00:15:02, lo0
C   2001:3::/64 [0/0]
    via ::, 00:14:55, gigabitethernet0
L   2001:3::1/128 [0/0]
    via ::, 00:14:54, lo0
```

According to the route table, Device1, Device2, and Device3 have learnt the routes of the loopback interfaces of each other.

Step 3: Configure the IPv6 BGP basic functions.

#Configure Device1.

```
Device1(config)#router bgp 100
Device1(config-bgp)#bgp router-id 1.1.1.1
Device1(config-bgp)#address-family ipv6
Device1(config-bgp-af)#neighbor 2::2 remote-as 100
Device1(config-bgp-af)#exit-address-family
Device1(config-bgp)#neighbor 2::2 update-source loopback 0
Device1(config-bgp)#exit
```

#Configure Device2.

```
Device2(config)#router bgp 100
Device2(config-bgp)#bgp router-id 2.2.2.2
Device2(config-bgp)#address-family ipv6
Device2(config-bgp-af)#neighbor 1::1 remote-as 100
Device2(config-bgp-af)#neighbor 3::3 remote-as 100
Device2(config-bgp-af)#exit-address-family
```

```
Device2(config-bgp)#neighbor 1::1 update-source loopback 0
Device2(config-bgp)#neighbor 3::3 update-source loopback 0
Device2(config-bgp)#exit
```

#Configure Device3.

```
Device3(config)#router bgp 100
Device3(config-bgp)#bgp router-id 3.3.3.3
Device3(config-bgp)#address-family ipv6
Device3(config-bgp-af)#neighbor 2::2 remote-as 100
Device3(config-bgp-af)#neighbor 2::2 next-hop-self
Device3(config-bgp-af)#neighbor 2001:3::2 remote-as 200
Device3(config-bgp-af)#exit-address-family
Device3(config-bgp)#neighbor 2::2 update-source loopback 0
Device3(config-bgp)#exit
```

#Configure Device4.

```
Device4#configure terminal
Device4(config)#router bgp 200
Device4(config-bgp)#bgp router-id 4.4.4.4
Device4(config-bgp)#address-family ipv6
Device4(config-bgp-af)#neighbor 2001:3::1 remote-as 100
Device4(config-bgp-af)#network 2001:4::/64
Device4(config-bgp-af)#exit-address-family
Device4(config-bgp)#exit
```

#On Device2, check the IPv6 BGP neighbor status.

```
Device2#show bgp ipv6 unicast summary
BGP router identifier 2.2.2.2, local AS number 100
BGP table version is 2
2 BGP AS-PATH entries
0 BGP community entries

Neighbor      V    AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down  State/PfxRcd
1::1          4   100      10      10        2    0    0 00:07:18        0
3::3          4   100      10       9        2    0    0 00:06:53        1

Total number of neighbors 2
```

#On Device4, check the IPv6 BGP neighbor status.

```
Device4#show bgp ipv6 unicast summary
BGP router identifier 4.4.4.4, local AS number 200
BGP table version is 2
1 BGP AS-PATH entries
0 BGP community entries

Neighbor        V    AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down  State/PfxRcd
2001:3::1       4   100       3       4        2    0    0 00:01:45        0

Total number of neighbors 1
```

According to the queried information, IPv6 BGP neighbors have been set up between the devices.

#Query the IPv6 BGP route table of Device3.

```
Device3#show bgp ipv6 unicast
BGP table version is 3, local router ID is 3.3.3.3
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
     Network          Next Hop          Metric     LocPrf Weight Path
[B]*> 2001:4::/64       2001:3::2           0               0 200 i
```

#View the IPv6 BGP route table of Device2.

```
Device2#show bgp ipv6 unicast
BGP table version is 7, local router ID is 2.2.2.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
     Network          Next Hop          Metric     LocPrf Weight Path
[B]*>i2001:4::/64       3::3                0        100      0 200 i
```

#Query the IPv6 BGP route table of Device1.

```
Device1#show bgp ipv6 unicast
```

According to the above result, Device2 and Device3 have learnt route 2001:4::/64, and Device2 has not advertised the route to Device1.

Step 4: Configure an IPv6 BGP route reflector.

#Configure Device2.

```
Device2(config)#router bgp 100
Device2(config-bgp)#address-family ipv6
Device2(config-bgp-af)#neighbor 1::1 route-reflector-client
Device2(config-bgp-af)#neighbor 3::3 route-reflector-client
Device2(config-bgp-af)#exit-address-family
Device2(config-bgp)#exit
```

On Device2, Device1 and Device3 have been configured as the RR clients.

Step 5: Check the result.

#View the route table of Device1.

```
Device1#show bgp ipv6 unicast
BGP table version is 2, local router ID is 1.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
      Network          Next Hop          Metric    LocPrf Weight Path
[B]*>i2001:4::/64       3::3              0       100      0 200 i
Device1#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management

L   ::1/128 [0/0]
     via ::, 1w2d:06:48:52, lo0
LC  1::1/128 [0/0]
     via ::, 00:36:32, loopback0
O   2::2/128 [110/2]
     via fe80::201:7aff:fec0:525a, 00:31:42, gigabitethernet0
O   3::3/128 [110/3]
     via fe80::201:7aff:fec0:525a, 00:23:12, gigabitethernet0
C   2001:1::/64 [0/0]
     via ::, 00:36:39, gigabitethernet0
L   2001:1::1/128 [0/0]
     via ::, 00:36:38, lo0
O   2001:2::/64 [110/2]
     via fe80::201:7aff:fec0:525a, 00:31:42, gigabitethernet0
B   2001:4::/64 [200/0]
     via 3::3, 00:01:16, gigabitethernet0
```

On BGP of Device2, Device1 and Device3 have been configured as the RR clients, and Device2 has successfully reflects route 2001:4::/64 to RR client Device1.

Note:

- If you configure an IPv6 BGP neighbor as a RR client, the neighbor will be reset.

12.3.5. Configure IPv6 BGP Route Summary

Network Requirements

- Set up OSPFv3 neighbors between Device1 and Device3, and configure Device3 to advertise routes 2002:1::/64 and 2002:2::/64 to Device1.
- Set up EBGP neighbors between Device1 and Device2.
- On Device1, aggregate routes 2002:1::/64 and 2002:2::/64 into route 2002::/30 and advertise the aggregated route to Device2.

Network Topology

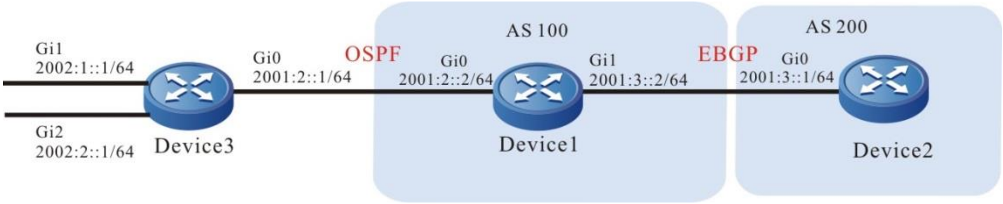

Figure 12–5 Networking for configuring IPv6 BGP route summary

Configuration Steps

Step 1: Configure the IPv6 global unicast addresses of the interfaces. (Omitted)

Step 2: Configure OSPFv3 so that Device1 can learn the two routes 2002:1:/64 and 2002:2::/64 advertised by Device3.

#Configure Device1.

```
Device1#configure terminal
Device1(config)#ipv6 router ospf 100
Device1(config-ospf6)#router-id 1.1.1.1
Device1(config-ospf6)#exit
Device1(config)#interface gigabitethernet 0
Device1(config-if-gigabitethernet0)#ipv6 router ospf 100 area 0
Device1(config-if-gigabitethernet0)#exit
```

#Configure Device3.

```
Device3#configure terminal
Device3(config)#ipv6 router ospf 100
Device3(config-ospf6)#router-id 3.3.3.3
Device3(config-ospf6)#exit
Device3(config)#interface gigabitethernet 0
Device3(config-if-gigabitethernet0)#ipv6 router ospf 100 area 0
Device3(config-if-gigabitethernet0)#exit
Device3(config)#interface gigabitethernet 1
Device3(config-if-gigabitethernet1)#ipv6 router ospf 100 area 0
Device3(config-if-gigabitethernet1)#exit
Device3(config)#interface gigabitethernet 2
```

```
Device3(config-if-gigabitethernet2)#ipv6 router ospf 100 area 0
Device3(config-if-gigabitethernet2)#exit
```

#View the route table of Device1.

```
Device1#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management

L   ::1/128 [0/0]
    via ::, 1w2d:07:35:38, lo0
C   2001:1::/64 [0/0]
    via ::, 00:01:11, gigabitethernet0
L   2001:1::2/128 [0/0]
    via ::, 00:01:10, lo0
C   2001:2::/64 [0/0]
    via ::, 00:01:06, gigabitethernet1
L   2001:2::2/128 [0/0]
    via ::, 00:01:04, lo0
O   2002:1::/64 [110/2]
    via fe80::201:7aff:fe62:bb7e, 00:01:54, gigabitethernet0
O   2002:2::/64 [110/2]
    via fe80::201:7aff:fe62:bb7e, 00:01:54, gigabitethernet0
```

According to the route table, Device1 has learnt routes 2002:1:1::/64 and 2002:1:2::/64 advertised by Device3.

Step 3: Configure the IPv6 BGP basic functions.

#Configure Device1.

```
Device1(config)#router bgp 100
Device1(config-bgp)#bgp router-id 1.1.1.1
Device1(config-bgp)#address-family ipv6
Device1(config-bgp-af)#neighbor 2001:2::1 remote-as 200
Device1(config-bgp-af)#exit-address-family
Device1(config-bgp)#exit
```

#Configure Device2.

```
Device2#configure terminal
Device2(config)#router bgp 200
Device2(config-bgp)#bgp router-id 2.2.2.2
Device2(config-bgp)#address-family ipv6
```

```
Device2(config-bgp-af)#neighbor 2001:2::2 remote-as 100
Device2(config-bgp-af)# xit-address-family
Device2(config-bgp)#exit
```

#On Device1, check the IPv6 BGP neighbor status.

```
Device1#show bgp ipv6 unicast summary
BGP router identifier 1.1.1.1, local AS number 100
BGP table version is 1
1 BGP AS-PATH entries
0 BGP community entries

Neighbor       V    AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down  State/PfxRcd
2001:2::1      4   200       3       3        1    0    0 00:01:16        0
```

IPv6 BGP neighbors have been successfully set up between Device1 and Device2.

Step 4: Configure IPv6 BGP route summary.

Two solutions are available to complete the network requirements.

Solution 1: Configure an IPv6 static route that is targeted at null0 to introduce the the static route to BGP.

#Configure Device1.

```
Device1(config)#ipv6 route 2002::/30 null 0
Device1(config)#router bgp 100
Device1(config-bgp)#address-family ipv6
Device1(config-bgp-af)#network 2002::/30
Device1(config-bgp-af)#exit-address-family
Device1(config-bgp)#exit
```

Check the result.

#View the IPv6 BGP route table of Device1.

```
Device1#show bgp ipv6 unicast
BGP table version is 2, local router ID is 1.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
      Network            Next Hop           Metric     LocPrf Weight Path
[B]*> 2002::/30          ::                  0            32768 i
```

You can see that the aggregated route 2001::/16 is generated in the IPv6 BGP route table of Device1.

#View the route table of Device2.

```
Device2#show bgp ipv6 unicast
```

```
BGP table version is 2, local router ID is 2.2.2.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
          S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
    Network            Next Hop          Metric    LocPrf Weight Path
[B]*> 2002::/30         2001:2::2             0            0 100 i
Device2#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
     U - Per-user Static route
     O - OSPF, OE-OSPF External, M - Management

L   ::1/128 [0/0]
    via ::, 1w6d:03:14:01, lo0
C   2001:2::/64 [0/0]
    via ::, 01:20:21, gigabitethernet1
L   2001:2::1/128 [0/0]
    via ::, 01:20:20, lo0
B   2002::/30 [20/0]
    via 2001:2::2, 00:00:44, gigabitethernet1
```

Device2 has successfully learnt the aggregated route 2002::/30 that has been advertised by Device1.

Solution 2: First introduce detailed routes into BGP, and then run the **aggregate-address** command to aggregate the routes.

#Configure Device1.

```
Device1(config)#router bgp 100
Device1(config-bgp)#address-family ipv6
Device1(config-bgp-af)#redistribute ospf 100
Device1(config-bgp-af)#aggregate-address 2002::/30 summary-only
Device1(config-bgp-af)#exit-address-family
Device1(config-bgp)#exit
```

Check the result.

#Query the IPv6 BGP route table of Device1.

```
Device1#show bgp ipv6 unicast
BGP table version is 4, local router ID is 1.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
          S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
    Network            Next Hop          Metric    LocPrf Weight Path
```

```
[O]*> 2001:1::/64       ::                 1          32768 ?
[B]*> 2002::/30         ::                            32768 i
[O]s> 2002:1::/64       ::                 2          32768 ?
[O]s> 2002:2::/64       ::                 2          32768 ?
```

The aggregated route 2002::/30 has been generated in the IPv6 BGP route table of Device1.

#View the route table of Device2.

```
Device2#show bgp ipv6 unicast
BGP table version is 4, local router ID is 2.2.2.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
    Network            Next Hop          Metric    LocPrf Weight Path
[B]*> 2001:1::/64       2001:2::2           1              0 100 ?
[B]*> 2002::/30         2001:2::2           0              0 100 i
Device2#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management

L    ::1/128 [0/0]
     via ::, 1w6d:03:16:42, lo0
B    2001:1::/64 [20/0]
     via 2001:2::2, 00:00:50, gigabitethernet0
C    2001:2::/64 [0/0]
     via ::, 01:23:01, gigabitethernet0
L    2001:2::1/128 [0/0]
     via ::, 01:23:00, lo0
B    2002::/30 [20/0]
     via 2001:2::2, 00:00:50, gigabitethernet0
```

Device2 has successfully learnt the aggregated route 2002::/30 that has been advertised by Device1.

Note:

- When the **aggregate-address** command is used to aggregate routes, if the extended command **summary-only** is configured, the device advertises only the aggregated route; otherwise, both common routes and aggregated routes are advertised.

12.3.6. Configure the IPv6 BGP Route Selection Priority

Network Requirements

- Set up IBGP neighbors between Device1 and Device2 and between Device1 and Device3, and set up EBGP neighbors between Device4 and Device2 and between Device4 and Device3.
- Device1 advertises two routes 2001:1::/64 and 2001:2::/64 to Device4, and Device4 advertises two routes 2001:7::/64 and 2001:8::/64 to Device1.
- Modify the Local-preference property of routes on Device2 and Device3 so that Device1 selects route 2001:7::/64 advertised by Device2 and route 2001:8::/64 advertised by Device3 with priority.
- Modify the MED property of routes on Device2 and Device3 so that Device4 selects route 2001:1::/64 advertised by Device3 and route 2001:2::/64 advertised by Device2 with priority.

Network Topology

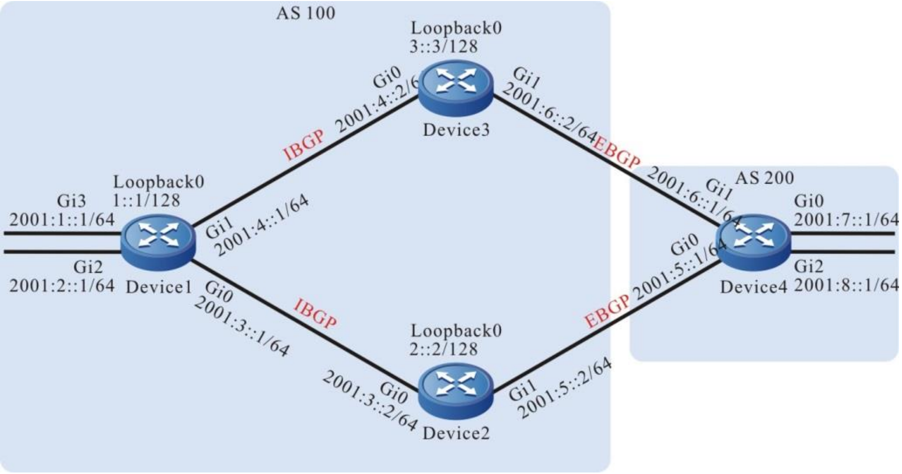

Figure 12–6 Networking for configuring the IPv6 BGP route selection priority

Configuration Steps

Step 1: Configure the IPv6 global unicast addresses of the interfaces. (Omitted)

Step 2: Configure OSPFv3 so that loopback routes are reachable between devices.

#Configure Device1.

```
Device1#configure terminal
Device1(config)#ipv6 router ospf 100
Device1(config-ospf6)#router-id 1.1.1.1
Device1(config-ospf6)#exit
Device1(config)#interface gigabitethernet 0
Device1(config-if-gigabitethernet0)#ipv6 router ospf 100 area 0
Device1(config-if-gigabitethernet0)#interface gigabitethernet 1
Device1(config-if-gigabitethernet1)#ipv6 router ospf 100 area 0
Device1(config-if-gigabitethernet1)#interface loopback 0
Device1(config-if-loopback0)#ipv6 router ospf 100 area 0
```

```
Device1(config-if-loopback0)#exit
```

#Configure Device2.

```
Device2#configure terminal
Device2(config)#ipv6 router ospf 100
Device2(config-ospf6)#router-id 2.2.2.2
Device2(config-ospf6)#exit
Device2(config)#interface gigabitethernet 0
Device2(config-if-gigabitethernet0)#ipv6 router ospf 100 area 0
Device2(config-if-gigabitethernet0)#exit
Device2(config)#interface loopback 0
Device2(config-if-loopback0)#ipv6 router ospf 100 area 0
Device2(config-if-loopback0)#exit
```

#Configure Device3.

```
Device3#configure terminal
Device3(config)#ipv6 router ospf 100
Device3(config-ospf6)#router-id 3.3.3.3
Device3(config-ospf6)#exit
Device3(config)#interface gigabitethernet 0
Device3(config-if-gigabitethernet0)#ipv6 router ospf 100 area 0
Device3(config-if-gigabitethernet0)#exit
Device3(config)#interface loopback 0
Device3(config-if-loopback0)#ipv6 router ospf 100 area 0
Device3(config-if-loopback0)#exit
```

#View the route table of Device1.

```
Device1#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
     U - Per-user Static route
     O - OSPF, OE-OSPF External, M - Management

L   ::1/128 [0/0]
    via ::, 1w5d:04:03:11, lo0
LC  1::1/128 [0/0]
    via ::, 00:08:39, loopback0
O   2::2/128 [110/2]
    via fe80::201:7aff:fe5e:87da, 00:02:04, gigabitethernet0
O   3::3/128 [110/2]
    via fe80::201:7aff:fec0:525b, 00:00:38, gigabitethernet1
```

```
C   2001:1::/64 [0/0]
     via ::, 00:09:12, gigabitethernet0/2/3
L   2001:1::1/128 [0/0]
     via ::, 00:09:11, lo0
C   2001:2::/64 [0/0]
     via ::, 00:08:26, gigabitethernet2
L   2001:2::1/128 [0/0]
     via ::, 00:08:26, gigabitethernet2
C   2001:3::/64 [0/0]
     via ::, 00:09:01, gigabitethernet0
L   2001:3::1/128 [0/0]
     via ::, 00:09:00, lo0
C   2001:4::/64 [0/0]
     via ::, 00:08:55, gigabitethernet1
L   2001:4::1/128 [0/0]
     via ::, 00:08:53, lo0
```

#View the route table of Device2.

```
Device2#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management

L   ::1/128 [0/0]
     via ::, 2w4d:23:16:51, lo0
O   1::1/128 [110/2]
     via fe80::201:7aff:fe62:bb7f, 00:04:25, gigabitethernet0
LC  2::2/128 [0/0]
     via ::, 00:09:31, loopback0
O   3::3/128 [110/3]
     via fe80::201:7aff:fe62:bb7f, 00:02:52, gigabitethernet0
C   2001:3::/64 [0/0]
     via ::, 00:09:49, gigabitethernet0
L   2001:3::2/128 [0/0]
     via ::, 00:09:48, lo0
O   2001:4::/64 [110/2]
     via fe80::201:7aff:fe62:bb7f, 00:04:25, gigabitethernet0
C   2001:5::/64 [0/0]
```

```
     via ::, 00:09:39, gigabitethernet1
L    2001:5::2/128 [0/0]
     via ::, 00:09:38, lo0
```

#View the route table of Device3.

```
Device3#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management

L    ::1/128 [0/0]
     via ::, 2w1d:22:16:55, lo0
O    1::1/128 [110/2]
     via fe80::201:7aff:fe62:bb80, 00:04:27, gigabitethernet0
O    2::2/128 [110/3]
     via fe80::201:7aff:fe62:bb80, 00:04:27, gigabitethernet0
LC  3::3/128 [0/0]
     via ::, 00:10:48, loopback0
O    2001:3::/64 [110/2]
     via fe80::201:7aff:fe62:bb80, 00:04:27, gigabitethernet0
C    2001:4::/64 [0/0]
     via ::, 00:11:55, gigabitethernet0
L    2001:4::2/128 [0/0]
     via ::, 00:11:54, lo0
C    2001:6::/64 [0/0]
     via ::, 00:11:48, gigabitethernet1
L    2001:6::2/128 [0/0]
     via ::, 00:11:47, lo0
```

According to the route table, Device1, Device2, and Device3 have learnt the routes of the loopback interfaces of each other.

Step 3: Configure the IPv6 BGP basic functions.

#Configure Device1.

```
Device1(config)#router bgp 100
Device1(config-bgp)#bgp router-id 1.1.1.1
Device1(config-bgp)#address-family ipv6
Device1(config-bgp-af)#neighbor 2::2 remote-as 100
Device1(config-bgp-af)#neighbor 3::3 remote-as 100
Device1(config-bgp-af)#network 2001:1::/64
```

```
Device1(config-bgp-af)#network 2001:2::/64
Device1(config-bgp-af)#exit-address-family
Device1(config-bgp)#neighbor 2::2 update-source loopback 0
Device1(config-bgp)#neighbor 3::3 update-source loopback 0
Device1(config-bgp)#exit
```

#Configure Device2.

```
Device2(config)#router bgp 100
Device2(config-bgp)#bgp router-id 2.2.2.2
Device2(config-bgp)#address-family ipv6
Device2(config-bgp-af)#neighbor 1::1 remote-as 100
Device2(config-bgp-af)#neighbor 1::1 next-hop-self
Device2(config-bgp-af)#neighbor 2001:5::1 remote-as 200
Device2(config-bgp-af)#exit-address-family
Device2(config-bgp)#neighbor 1::1 update-source loopback 0
Device2(config-bgp)#exit
```

#Configure Device3.

```
Device3(config)#router bgp 100
Device3(config-bgp)#bgp router-id 3.3.3.3
Device3(config-bgp)#address-family ipv6
Device3(config-bgp-af)#neighbor 1::1 remote-as 100
Device3(config-bgp-af)#neighbor 1::1 next-hop-self
Device3(config-bgp-af)#neighbor 2001:6::1 remote-as 200
Device3(config-bgp-af)#exit-address-family
Device3(config-bgp)#neighbor 1::1 update-source loopback 0
Device3(config-bgp)#exit
```

#Configure Device4.

```
Device4#configure terminal
Device4(config)#router bgp 200
Device4(config-bgp)#bgp router-id 4.4.4.4
Device4(config-bgp)#address-family ipv6
Device4(config-bgp-af)#neighbor 2001:5::2 remote-as 100
Device4(config-bgp-af)#neighbor 2001:6::2 remote-as 100
Device4(config-bgp-af)#network 2001:7::/64
Device4(config-bgp-af)#network 2001:8::/64
Device4(config-bgp-af)#exit-address-family
Device4(config-bgp)#exit
```

#On Device1, check the IPv6 BGP neighbor status.

```
Device1#show bgp ipv6 unicast summary
BGP router identifier 1.1.1.1, local AS number 100
BGP table version is 4
2 BGP AS-PATH entries
0 BGP community entries

Neighbor        V    AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down  State/PfxRcd
2::2            4   100       9      10        4    0    0 00:06:18        2
3::3            4   100       7       8        4    0    0 00:04:29        2

Total number of neighbors 2
```

#On Device4, check the IPv6 BGP neighbor status.

```
Device4#show bgp ipv6 unicast summary
BGP router identifier 4.4.4.4, local AS number 200
BGP table version is 4
2 BGP AS-PATH entries
0 BGP community entries

Neighbor        V    AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down  State/PfxRcd
2001:5::2       4   100       6       5        4    0    0 00:02:43        2
2001:6::2       4   100       5       6        4    0    0 00:02:32        2

Total number of neighbors 2
```

IBGP neighbors have been set up between Device1 and Device2 and between Device2 and Device3, and EBGP neighbors have been set up between Device4 and Device2 and between Device4 and Device3.

#View the route table of Device1.

```
Device1#show bgp ipv6 unicast
BGP table version is 4, local router ID is 1.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
          S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
    Network          Next Hop          Metric     LocPrf Weight Path
[B]*> 2001:1::/64      ::                 0          32768 i
[B]*> 2001:2::/64      ::                 0          32768 i
[B]* i2001:7::/64      3::3                0       100      0 200 i
[B]*>i               2::2                0       100      0 200 i
[B]*>i2001:8::/64      2::2                0       100      0 200 i
```

```
[B]* i              3::3               0       100      0 200 i

Device1#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
     U - Per-user Static route
     O - OSPF, OE-OSPF External, M - Management

L   ::1/128 [0/0]
    via ::, 1w5d:04:20:19, lo0
LC  1::1/128 [0/0]
    via ::, 00:25:47, loopback0
O   2::2/128 [110/2]
    via fe80::201:7aff:fe5e:87da, 00:19:12, gigabitethernet0
O   3::3/128 [110/2]
    via fe80::201:7aff:fec0:525b, 00:17:46, gigabitethernet1
C   2001:1::/64 [0/0]
    via ::, 00:26:20, gigabitethernet0/2/3
L   2001:1::1/128 [0/0]
    via ::, 00:26:19, lo0
C   2001:2::/64 [0/0]
    via ::, 00:25:34, gigabitethernet2
L   2001:2::1/128 [0/0]
    via ::, 00:25:34, gigabitethernet2
C   2001:3::/64 [0/0]
    via ::, 00:26:09, gigabitethernet0
L   2001:3::1/128 [0/0]
    via ::, 00:26:08, lo0
C   2001:4::/64 [0/0]
    via ::, 00:26:03, gigabitethernet1
L   2001:4::1/128 [0/0]
    via ::, 00:26:01, lo0
B   2001:7::/64 [200/0]
    via 2::2, 00:03:21, gigabitethernet0
B   2001:8::/64 [200/0]
    via 2::2, 00:02:57, gigabitethernet0
```

According to the route table, both route 2001:7::/64 and route 2001:8::/64 of Device1 select Device2 as the next-hop device.

#Query the route table of Device4.

```
Device4#show bgp ipv6 unicast
BGP table version is 4, local router ID is 4.4.4.4
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
          S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
    Network          Next Hop          Metric     LocPrf Weight Path
[B]*  2001:1::/64       2001:6::2           0              0 100 i
[B]*>                 2001:5::2           0              0 100 i
[B]*  2001:2::/64       2001:6::2           0              0 100 i
[B]*>                 2001:5::2           0              0 100 i
[B]*> 2001:7::/64       ::                  0          32768 i
[B]*> 2001:8::/64       ::                  0          32768 i

Device4#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
     U - Per-user Static route
     O - OSPF, OE-OSPF External, M - Management



L   ::1/128 [0/0]
    via ::, 1w5d:04:14:15, lo0
B   2001:1::/64 [20/0]
    via 2001:5::2, 00:06:52, gigabitethernet0
B   2001:2::/64 [20/0]
    via 2001:5::2, 00:06:52, gigabitethernet0
C   2001:5::/64 [0/0]
    via ::, 00:26:17, gigabitethernet0
L   2001:5::1/128 [0/0]
    via ::, 00:26:16, lo0
C   2001:6::/64 [0/0]
    via ::, 00:26:24, gigabitethernet1
L   2001:6::1/128 [0/0]
    via ::, 00:26:23, lo0
C   2001:7::/64 [0/0]
    via ::, 00:25:53, gigabitethernet0/2/3
L   2001:7::1/128 [0/0]
```

```
     via ::, 00:25:51, lo0
C    2001:8::/64 [0/0]
     via ::, 00:25:40, gigabitethernet2
L    2001:8::1/128 [0/0]
     via ::, 00:25:40, gigabitethernet2
```

Both route 2001:1::/64 and 2001:2::/64 of Device4 select Device3 as the next-hop device.

Step 4: Configure an ACL and routing policy to set local-preference and metric.

#Configure Device2.

```
Device2(config)#ipv6 access-list extended 7001
Device2(config-v6-list)#permit ipv6 2001:8::/64 any
Device2(config-v6-list)#exit
Device2(config)#ipv6 access-list extended 7002
Device2(config-v6-list)#permit ipv6 2001:1::/64 any
Device2(config-v6-list)#exit
Device2(config)#route-map SetPriority1 10
Device2(config-route-map)#match ipv6 address 7001
Device2(config-route-map)#set local-preference 110
Device2(config-route-map)#exit
Device2(config)#route-map SetPriority1 20
Device2(config-route-map)#set local-preference 20
Device2(config-route-map)#exit
Device2(config)#route-map SetPriority2 10
Device2(config-route-map)#match ipv6 address 7002
Device2(config-route-map)#set metric 100
Device2(config-route-map)#exit
Device2(config)#route-map SetPriority2 20
Device2(config-route-map)#set metric 20
Device2(config-route-map)#exit
```

On Device2, configure a routing policy to set local-preference of route 2001:8::/64 to 110, and set metric of route 2001:1::/64 to 100.

#Configure Device3.

```
Device3(config)#ipv6 access-list extended 7001
Device3(config-v6-list)#permit ipv6 2001:7::/64 any
Device3(config-v6-list)#exit
Device3(config)#ipv6 access-list extended 7002
Device3(config-v6-list)#permit ipv6 2001:2::/64 any
Device3(config-v6-list)#exit
```

```
Device3(config)#route-map SetPriority1 10
Device3(config-route-map)#match ipv6 address 7001
Device3(config-route-map)#set local-preference 110
Device3(config-route-map)#exit
Device3(config)#route-map SetPriority1 20
Device3(config-route-map)#exit
Device3(config)#route-map SetPriority2 10
Device3(config-route-map)#match ipv6 address 7002
Device3(config-route-map)#set metric 100
Device3(config-route-map)#exit
Device3(config)#route-map SetPriority2 20
Device3(config-route-map)#exit
```

On Device3, configure a routing policy to set local-preference of route 2001:7::/64 to 110, and set metric of route 2001:2::/64 to 100.

Note:

- In configuring a routing policy, you can create a filtration rule based on a prefix list or ACL. The prefix list can precisely match routing masks while the ACL cannot match routing masks.

Step 5: Configure a routing policy for IPv6 BGP.

#Configure Device2.

```
Device2(config)#router bgp 100
Device2(config-bgp)#address-family ipv6
Device2(config-bgp-af)#neighbor 1::1 route-map SetPriority1 out
Device2(config-bgp-af)#neighbor 2001:5::1 route-map SetPriority2 out
Device2(config-bgp-af)#exit-address-family
Device2(config-bgp)#exit
```

On Device2, configure the outgoing direction of neighbor 1::1 to modify local-preference of route 2001:8::/64, and configure the outgoing direction of neighbor 2001:5::1 to modify metric of route 2001:1::/64.

#Configure Device3.

```
Device3(config)#router bgp 100
Device3(config-bgp)#address-family ipv6
Device3(config-bgp-af)#neighbor 1::1 route-map SetPriority1 out
Device3(config-bgp-af)#neighbor 2001:6::1 route-map SetPriority2 out
Device3(config-bgp-af)# exit-address-family
Device2(config-bgp)#exit
```

On Device3, configure the outgoing direction of neighbor 10.0.0.1 to modify local-preference of route 2001:7::/64, and configure the outgoing direction of neighbor 2001:6::1 to modify metric of route2001:2::/64.

After a routing policy is configured on the neighbor, you need to reset the neighbor.

Step 6: Check the result.

#View the route table of Device1.

```
Device1#show bgp ipv6 unicast
BGP table version is 9, local router ID is 1.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
          S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
    Network           Next Hop          Metric    LocPrf Weight Path
[B]*> 2001:1::/64       ::                 0          32768 i
[B]*> 2001:2::/64       ::                 0          32768 i
[B]* i2001:7::/64       2::2               0       100      0 200 i
[B]*>i                3::3               0       110      0 200 i
[B]*>i2001:8::/64       2::2               0       110      0 200 i
[B]* i                3::3               0       100      0 200 i

Device1#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
     U - Per-user Static route
     O - OSPF, OE-OSPF External, M - Management

L   ::1/128 [0/0]
    via ::, 1w5d:04:59:59, lo0
LC  1::1/128 [0/0]
    via ::, 01:05:27, loopback0
O   2::2/128 [110/2]
    via fe80::201:7aff:fe5e:87da, 00:58:52, gigabitethernet1
O   3::3/128 [110/2]
    via fe80::201:7aff:fec0:525b, 00:57:26, gigabitethernet2
C   2001:1::/64 [0/0]
    via ::, 01:06:00, gigabitethernet0
L   2001:1::1/128 [0/0]
    via ::, 01:05:59, lo0
C   2001:2::/64 [0/0]
    via ::, 01:05:14, loopback1
L   2001:2::1/128 [0/0]
    via ::, 01:05:14, loopback1
```

```
C   2001:3::/64 [0/0]
    via ::, 01:05:49, gigabitethernet1
L   2001:3::1/128 [0/0]
    via ::, 01:05:48, lo0
C   2001:4::/64 [0/0]
    via ::, 01:05:43, gigabitethernet2
L   2001:4::1/128 [0/0]
    via ::, 01:05:41, lo0
B   2001:7::/64 [200/0]
    via 3::3, 00:09:05, gigabitethernet1
B   2001:8::/64 [200/0]
    via 2::2, 00:04:58, gigabitethernet0
```

According to the route table, local-preference of routes 2001:7::/64 and 2001:8::/64 is modified successfully, and Device1 select route 2001:8::/64 that is advertised by Device2 and route2001:7::/64 that is advertised by Device3 with priority.

#Query the route table of Device4.

```
Device4#show bgp ipv6 unicast
BGP table version is 5, local router ID is 4.4.4.4
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
           S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
     Network            Next Hop          Metric     LocPrf Weight Path
[B]*  2001:1::/64        2001:5::2          100              0 100 i
[B]*>                  2001:6::2          0              0 100 i
[B]*> 2001:2::/64        2001:5::2            0              0 100 i
[B]*                   2001:6::2         100              0 100 i
[B]*> 2001:7::/64        ::                  0          32768 i
[B]*> 2001:8::/64        ::                  0          32768 i

Device4#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
      U - Per-user Static route
      O - OSPF, OE-OSPF External, M - Management

L   ::1/128 [0/0]
    via ::, 1w5d:04:53:45, lo0
B   2001:1::/64 [20/0]
    via 2001:6::2, 00:12:10, gigabitethernet1
```

```
B   2001:2::/64 [20/0]
    via 2001:5::2, 00:07:40, gigabitethernet0
C   2001:5::/64 [0/0]
    via ::, 01:05:47, gigabitethernet0
L   2001:5::1/128 [0/0]
    via ::, 01:05:46, lo0
C   2001:6::/64 [0/0]
    via ::, 01:05:54, gigabitethernet1
L   2001:6::1/128 [0/0]
    via ::, 01:05:52, lo0
C   2001:7::/64 [0/0]
    via ::, 01:05:22, gigabitethernet0/2/3
L   2001:7::1/128 [0/0]
    via ::, 01:05:21, lo0
C   2001:8::/64 [0/0]
    via ::, 01:05:09, gigabitethernet2
L   2001:8::1/128 [0/0]
    via ::, 01:05:09, gigabitethernet2
```

According to the route table, metric of routes 2001:1::/64 and 2001:2::/64 is modified successfully, and Device4 select route 2001:2::/64 that is advertised by Device2 and route 2001:1::/64 that is advertised by Device3 with priority.

Note:

- A routing policy can be used in the outgoing direction of route advertisement, and it can also be used in the incoming direction of route receiving.

12.3.7. Configure IPv6 BGP to Coordinate with BFD

Network Requirements

- Set up EBGP neighbors between Device1 and Device2 and between Device1 and Device3, and set up IBGP neighbors between Device2 and Device3.
- Device1 learns EBGP route 2001:3::/64 both from Device2 and Device3, and Device1 selects to forward data to the network segment 2001:3::/64 through Device2.
- On Device1 and Device2, configure EBGP to coordinate with BFD. When the line between Device1 and Device2 becomes faulty, BFD can quickly detect the fault and notify BGP of the fault. Then Device1 selects to forward data to network segment 2001:3::/64 through Device3.

Network Topology

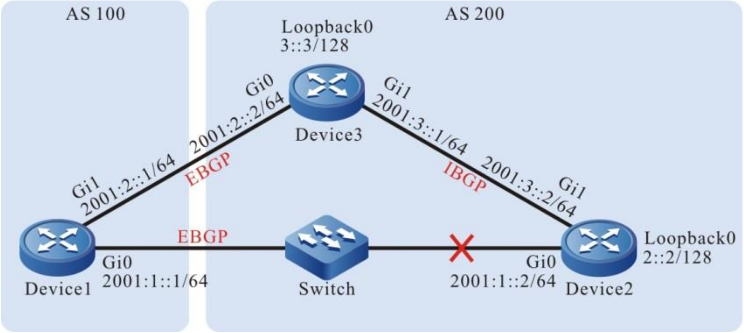

Figure 12–7 Networking for configuring IPv6 BGP to coordinate with BFD

Configuration Steps

Step 1: Configure the IPv6 global unicast addresses of the interfaces. (Omitted)

Step 2: Configure OSPFv3 so that loopback routes are reachable between devices.

#Configure Device2.

```
Device2#configure terminal
Device2(config)#ipv6 router ospf 100
Device2(config-ospf6)#router-id 2.2.2.2
Device2(config-ospf6)#exit
Device2(config)# interface gigabitethernet 1
Device2(config-if-gigabitethernet1)#ipv6 router ospf 100 area 0
Device2(config-if-gigabitethernet1)#exit
Device2(config)#interface loopback 0
Device2(config-if-loopback0)#ipv6 router ospf 100 area 0
Device2(config-if-loopback0)#exit
```

#Configure Device3.

```
Device3#configure terminal
Device3(config)#ipv6 router ospf 100
Device3(config-ospf6)#router-id 3.3.3.3
Device3(config-ospf6)#exit
Device3(config)#interface gigabitethernet 1
Device3(config-if-gigabitethernet1)#ipv6 router ospf 100 area 0
Device3(config-if-gigabitethernet1)#exit
Device3(config)# interface loopback 0
Device3(config-if-loopback0)#ipv6 router ospf 100 area 0
Device3(config-if-loopback0)#exit
```

#View the route table of Device2.

```
Device2#show ipv6 route
```

```
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management

L   ::1/128 [0/0]
     via ::, 1w2d:09:31:22, lo0
LC  2::2/128 [0/0]
     via ::, 00:10:10, loopback0
O   3::3/128 [110/1]
     via fe80::201:7aff:fec0:525a, 00:00:12, gigabitethernet1
C   2001:1::/64 [0/0]
     via ::, 00:10:54, gigabitethernet0
L   2001:1::2/128 [0/0]
     via ::, 00:10:53, lo0
C   2001:3::/64 [0/0]
     via ::, 00:10:17, gigabitethernet1
L   2001:3::2/128 [0/0]
     via ::, 00:10:16, lo0
```

#View the route table of Device3.

```
Device3#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management

L   ::1/128 [0/0]
     via ::, 1w6d:03:50:38, lo0
O   2::2/128 [110/2]
     via fe80::201:7aff:fe5e:6d2e, 00:02:40, gigabitethernet1
LC  3::3/128 [0/0]
     via ::, 00:00:49, loopback0
C   2001:2::/64 [0/0]
     via ::, 00:03:03, gigabitethernet0
L   2001:2::2/128 [0/0]
     via ::, 00:03:02, lo0
C   2001:3::/64 [0/0]
     via ::, 00:03:18, gigabitethernet1
L   2001:3::1/128 [0/0]
```

```
      via ::, 00:03:17, lo0
```

According to the route table, Device2 and Device3 have learnt the routes of the loopback interfaces of each other.

Step 3: Configure an ACL and routing policy to set the metric of a route.

#Configure Device1.

```
Device1#configure terminal
Device1(config)#ipv6 access-list extended 7001
Device1(config-v6-list)#permit ipv6 2001:3::/64 any
Device1(config-v6-list)#exit
Device1(config)#route-map SetMetric
Device1(config-route-map)#match ipv6 address 7001
Device1(config-route-map)#set metric 50
Device1(config-route-map)#exit
```

The routing policy that is configured on Device1 sets the metric of route 2001:3::/64 to 50.

Step 4: Configure the IPv6 BGP basic functions, and associate Device1 the routing policy.

#Configure Device1.

```
Device1(config)#router bgp 100
Device1(config-bgp)#bgp router-id 1.1.1.1
Device1(config-bgp)#address-family ipv6
Device1(config-bgp-af)#neighbor 2001:1::2 remote-as 200
Device1(config-bgp-af)#neighbor 2001:2::2 remote-as 200
Device1(config-bgp-af)#neighbor 2001:2::2 route-map SetMetric in
Device1(config-bgp-af)#exit-address-family
Device1(config-bgp)#exit
```

#Configure Device2.

```
Device2(config)#router bgp 200
Device2(config-bgp)#bgp router-id 2.2.2.2
Device2(config-bgp)#address-family ipv6
Device2(config-bgp-af)#neighbor 2001:1::1 remote-as 100
Device2(config-bgp-af)#neighbor 3::3 remote-as 200
Device2(config-bgp-af)#network 2001:3::/64
Device2(config-bgp-af)#exit-address-family
Device2(config-bgp)#neighbor 3::3 update-source loopback 0
Device2(config-bgp)#exit
```

#Configure Device3.

```
Device3(config)#router bgp 200
```

```
Device3(config-bgp)#bgp router-id 3.3.3.3
Device3(config-bgp)#address-family ipv6
Device3(config-bgp-af)#neighbor 2001:2::1 remote-as 100
Device3(config-bgp-af)#neighbor 2::2 remote-as 200
Device3(config-bgp-af)#network 2001:3::/64
Device3(config-bgp-af)#exit-address-family
Device3(config-bgp)#neighbor 2::2 update-source loopback 0
Device3(config-bgp)#exit
```

After a routing policy is configured on the peer, you need to reset the peer.

#On Device1, check the IPv6 BGP neighbor status.

```
Device1#show bgp ipv6 unicast summary
BGP router identifier 1.1.1.1, local AS number 100
BGP table version is 2
2 BGP AS-PATH entries
0 BGP community entries

Neighbor        V    AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down  State/PfxRcd
2001:1::2       4   200       7       6        2    0    0 00:04:00        1
2001:2::2       4   200       5       5        2    0    0 00:02:03        1

Total number of neighbors 2
```

#On Device2, check the IPv6 BGP neighbor status.

```
Device2#show bgp ipv6 unicast summary
BGP router identifier 2.2.2.2, local AS number 200
BGP table version is 2
1 BGP AS-PATH entries
0 BGP community entries

Neighbor        V    AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down  State/PfxRcd
3::3            4   200       5       5        2    0    0 00:02:10        1
2001:1::1       4   100       6       7        2    0    0 00:04:38        0

Total number of neighbors 2
```

IPv6 BGP neighbors between Device1, Device2, and Device3 have been set up successfully.

```
Device1#show bgp ipv6 unicast
BGP table version is 2, local router ID is 1.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
     Network           Next Hop         Metric     LocPrf Weight Path
[B]*  2001:3::/64       2001:2::2          50              0 200 i
[B]*>                 2001:1::2           0              0 200 i

Device1#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management

L   ::1/128 [0/0]
    via ::, 1w2d:09:53:27, lo0
C   2001:1::/64 [0/0]
    via ::, 00:24:05, gigabitethernet0
L   2001:1::1/128 [0/0]
    via ::, 00:24:02, lo0
C   2001:2::/64 [0/0]
    via ::, 00:25:21, gigabitethernet1
L   2001:2::1/128 [0/0]
    via ::, 00:25:20, lo0
B   2001:3::/64 [20/0]
    via 2001:1::2, 00:05:06, gigabitethernet0
```

According to the route table, route 2001:3::/64 of Device1 selects Device2 as the next-hop device.

Step 5: Configure IPv6 BGP to link with BFD.

#Configure Device1.

```
Device1(config)#router bgp 100
Device1(config-bgp)#address-family ipv6
Device1(config-bgp-af)#neighbor 2001:1::2 fall-over bfd
Device1(config-bgp-af)#exit-address-family
Device1(config-bgp)#exit
Device1(config)#interface gigabitethernet 0
Device1(config-if-gigabitethernet0)#bfd min-transmit-interval 500
Device1(config-if-gigabitethernet0)#bfd min-receive-interval 500
Device1(config-if-gigabitethernet0)#bfd multiplier 4
Device1(config-if-gigabitethernet0)#exit
```

#Configure Device2.

```
Device2(config)#router bgp 200
Device2(config-bgp)#address-family ipv6
Device2(config-bgp-af)#neighbor 2001:1::1 fall-over bfd
Device2(config-bgp-af)#exit-address-family
Device2(config-bgp)#exit
Device2(config)#interface gigabitethernet 0
Device2(config-if-gigabitethernet0)#bfd min-transmit-interval 500
Device2(config-if-gigabitethernet0)#bfd min-receive-interval 500
Device2(config-if-gigabitethernet0)#bfd multiplier 4
Device2(config-if-gigabitethernet0)#exit
```

BFD is enabled between EBGP neighbors Device1 and Device2, and the minimum transmit interval, minimum receive interval, and detection timeout multiple of the BFD control packets have been modified.

Step 6: Check the result.

#On Device1, query the BFD session status.

```
Device1#show bfd session ipv6
Device1#show bfd session ipv6
OurAddr                      NeighAddr          LD/RD           State      Holddown
Interface
2001:1::1                      2001:1::2          1026/2021         UP          2000
gigabitethernet0
```

On Device1, the BFD status is up, and the holddown time is negotiated to be 2000ms.

#If the line between Device1 and Device2 becomes faulty, the route can quickly switch over to the backup line.

#View the route table of Device1.

```
Device1#show bgp ipv6 unicast
BGP table version is 3, local router ID is 1.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
         S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
    Network          Next Hop          Metric    LocPrf Weight Path
[B]*> 2001:3::/64        2001:2::2          50             0 200 i
Device1#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
     U - Per-user Static route
     O - OSPF, OE-OSPF External, M - Management
```

```
L   ::1/128 [0/0]
    via ::, 1w2d:10:06:30, lo0
C   2001:1::/64 [0/0]
    via ::, 00:37:08, gigabitethernet0
L   2001:1::1/128 [0/0]
    via ::, 00:37:04, lo0
C   2001:2::/64 [0/0]
    via ::, 00:38:24, gigabitethernet1
L   2001:2::1/128 [0/0]
    via ::, 00:38:23, lo0
B   2001:3::/64 [20/0]
    via 2001:2::2, 00:00:16, gigabitethernet1
```

The next hop of route 2001:3::/64 is Device3.

12.3.8. Configure IPv6 BGP Authentication

Network Requirements

- Set up EBGP neighbors between Device1 and Device2, and set up IBGP neighbors between Device2 and Device3.
- Device1 learns the interface direct route 2001:4::/64 of Device3, and Device3 learns the interface direct route 2001:1::/64 of Device1.

Network Topology

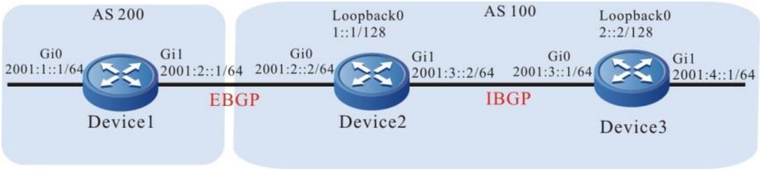

Figure 12–8 Networking for configuring IPv6 BGP authentication

Configuration Steps

Step 1: Configure the IPv6 global unicast addresses of the interfaces. (Omitted)

Step 2: Configure OSPFv3 so that loopback routes are reachable between devices.

#Configure Device2.

```
Device2#configure terminal
Device2(config)#ipv6 router ospf 100
Device2(config-ospf6)#router-id 2.2.2.2
Device2(config-ospf6)#exit
Device2(config)#interface gigabitethernet 1
Device2(config-if-gigabitethernet1)#ipv6 router ospf 100 area 0
Device2(config-if-gigabitethernet1)#exit
```

```
Device2(config)#interface loopback 0
Device2(config-if-loopback0)#ipv6 router ospf 100 area 0
Device2(config-if-loopback0)#exit
```

#Configure Device3.

```
Device3#configure terminal
Device3(config)#ipv6 router ospf 100
Device3(config-ospf6)#router-id 3.3.3.3
Device3(config-ospf6)#exit
Device3(config)#interface gigabitethernet 0
Device3(config-if-gigabitethernet0)#ipv6 router ospf 100 area 0
Device3(config-if-gigabitethernet0)#exit
Device3(config)#interface loopback 0
Device3(config-if-loopback0)#ipv6 router ospf 100 area 0
Device3(config-if-loopback0)#exit
```

#View the route table of Device2.

```
Device2#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
     U - Per-user Static route
     O - OSPF, OE-OSPF External, M - Management

L   ::1/128 [0/0]
    via ::, 1w1d:23:51:37, lo0
LC  1::1/128 [0/0]
    via ::, 00:09:34, loopback0
O   2::2/128 [110/2]
    via fe80::201:7aff:fec0:525a, 00:05:29, gigabitethernet1
C   2001:2::/64 [0/0]
    via ::, 00:09:41, gigabitethernet0
L   2001:2::2/128 [0/0]
    via ::, 00:09:39, lo0
C   2001:3::/64 [0/0]
    via ::, 00:08:55, gigabitethernet1
L   2001:3::2/128 [0/0]
    via ::, 00:08:53, lo0
```

#View the route table of Device3.

```
Device3#show ipv6 route
```

```
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management

L   ::1/128 [0/0]
    via ::, 1w5d:18:34:53, lo0
O   1::1/128 [110/2]
    via fe80::201:7aff:fe5e:6d2e, 00:29:59, gigabitethernet0
LC  2::2/128 [0/0]
    via ::, 00:32:36, loopback0
C   2001:3::/64 [0/0]
    via ::, 00:32:59, gigabitethernet0
L   2001:3::1/128 [0/0]
    via ::, 00:32:58, lo0
C   2001:4::/64 [0/0]
    via ::, 00:32:44, gigabitethernet1
L   2001:4::1/128 [0/0]
    via ::, 00:32:43, lo0
```

According to the queried information, Device2 and Device3 have learnt the routes of the peer loopback interfaces by running OSPFv3, preparing for setting up IBGP neighbors on the loopback interfaces of Device2 and Device3.

Step 3: Configure the IPv6 BGP basic functions.

#Configure Device1.

Set up a direct-connect EBGP peer with Device2. Introduce 2001:1::/64 to BGP in network mode.

```
Device1#configure terminal
Device1(config)#router bgp 200
Device1(config-bgp)#bgp router-id 1.1.1.1
Device1(config-bgp)#address-family ipv6
Device1(config-bgp-af)#neighbor 2001:2::2 remote-as 100
Device1(config-bgp-af)#network 2001:1::/64
Device1(config-bgp-af)#exit-address-family
Device1(config-bgp)#exit
```

#Configure Device2.

Set up the direct-connect EBGP peer with Device1, set up a non-direct-connect IBGP peer with Device3 through Loopback0, and set the next hop of the advertised route to the local device.

```
Device2(config)#router bgp 100
Device2(config-bgp)#bgp router-id 2.2.2.2
```

```
Device2(config-bgp)#address-family ipv6
Device2(config-bgp-af)#neighbor 2001:2::1 remote-as 200
Device2(config-bgp-af)#neighbor 2::2 remote-as 100
Device2(config-bgp-af)#neighbor 2::2 next-hop-self
Device2(config-bgp-af)#exit-address-family
Device2(config-bgp)#neighbor 2::2 update-source loopback 0
Device2(config-bgp)#exit
```

#Configure Device3.

Set up a non-direct-connect IBGP peer relation with Device2 through Loopback0. Introduce 2001:4::/64 to BGP in network mode.

```
Device3(config)#router bgp 100
Device3(config-bgp)#bgp router-id 3.3.3.3
Device3(config-bgp)#address-family ipv6
Device3(config-bgp-af)#neighbor 1::1 remote-as 100
Device3(config-bgp-af)#network 2001:4::/64
Device3(config-bgp-af)#exit-address-family
Device3(config-bgp)#neighbor 1::1 update-source loopback 0
Device3(config-bgp)#exit
```

#View the IPv6 BGP neighbor status on Device2.

```
Device2#show bgp ipv6 unicast summary
BGP router identifier 2.2.2.2, local AS number 100
BGP table version is 4
2 BGP AS-PATH entries
0 BGP community entries

Neighbor        V    AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down  State/PfxRcd
2::2            4   100      8       6        3    0    0 00:04:12        1
2001:2::1       4   200     15      15        3    0    0 00:11:17        1

Total number of neighbors 2
```

According to the numbers (Number of route prefixes received from neighbors) that are displayed in the State/PfxRcd column, IPv6 BGP neighbors have been successfully set up between Device 2 and Device 1, Device 3.

Step 4: Configure the IPv6 BGP neighbor authentication.

#Configure Device1, setting the authentication password admin1 for Device2.

```
Device1#configure terminal
```

```
Device1(config)#router bgp 200
Device1(config-bgp)#address-family ipv6
Device1(config-bgp-af)#neighbor 2001:2::2 password 0 admin1
Device1(config-bgp-af)#exit-address-family
Device1(config-bgp)#exit
```

#Configure Device3, setting the authentication password for Device2 as admin2.

```
Device3(config)#router bgp 100
Device3(config-bgp)#address-family ipv6
Device3(config-bgp-af)#neighbor 1::1 password 0 admin2
Device3(config-bgp-af)#exit-address-family
Device3(config-bgp)#exit
```

#View the IPv6 BGP neighbor status on Device2.

```
Device2#show bgp ipv6 unicast summary
BGP router identifier 2.2.2.2, local AS number 100
BGP table version is 4
2 BGP AS-PATH entries
0 BGP community entries

Neighbor      V   AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down  State/PfxRcd
2::2          4  100      0       0       0    0    0   never Connect
2001:2::1     4  200      0       0       0    0    0   never Connect

Total number of neighbors 2
```

According to the content **Connect** displayed in the State/PfxRcd column, the IPv6 BGP neighbors are not set up between Device2 and Device1, Device3.

#Configure Device2, setting the authentication passwords of Device1 and Device3 to admin1 and admin2 respectively.

```
Device2(config)#router bgp 100
Device2(config-bgp)#address-family ipv6
Device2(config-bgp-af)#neighbor 2001:2::1 password 0 admin1
Device2(config-bgp-af)#neighbor 2::2 password 0 admin2
Device2(config-bgp-af)#exit-address-family
Device2(config-bgp)#exit
```

Step 5: Check the result.

#View the IPv6 BGP neighbor status on Device2.

```
Device2#show bgp ipv6 unicast summary
BGP router identifier 2.2.2.2, local AS number 100
BGP table version is 4
2 BGP AS-PATH entries
0 BGP community entries

Neighbor        V    AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down  State/PfxRcd
2::2          4   100       6       6        3    0    0 00:06:12        1
2001:2::1       4   200      12      10        3    0    0 00:09:15        1

Total number of neighbors 2
```

According to the numbers (Number of route prefixes received from neighbors) that are displayed in the State/PfxRcd column, IPv6 BGP neighbors have been successfully set up between Device 2 and Device 1, Device 3.

12.3.9. Configure IPv6 BGP to Use IPSec Encryption Authentication

Network Requirements

- EBGP neighbors are established between Device1 and Device2 through neighbor authentication, and IBGP neighbors are established between Device2 and Device3 through neighbor authentication.
- Device1, Device2 and Device3 use the IPSec tunnel to encrypt and authenticate IPv6 BGP protocol packets. Device1 and Device2 adopt ESP transmission encapsulation, with encryption algorithm of 3DES and authentication algorithm of SHA1. Device2 and Device3 adopt ESP transmission encapsulation, with encryption algorithm of aes128 and ESP authentication algorithm of sm3.
- Device1 learns the interface direct connection route 2001:4::/64 of Device3, and Device3 learns the interface direct connection route 2001:1::/64 of Device1.

Network Topology

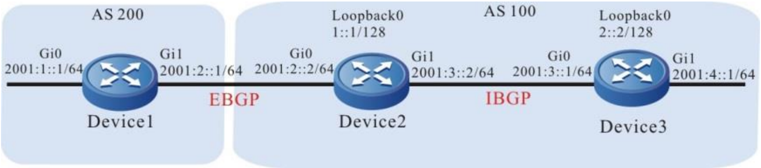

Figure 12-9 Networking of configuring IPv6 BGP to use IPSec encryption authentication

Configuration Steps

Step 1: Configure the IPv6 global unicast address of the interface (omitted).

Step 2: Configure OSPFv3, making the loopback interface route between devices reachable.

#Configure Device2.

```
Device2#configure terminal
Device2(config)#ipv6 router ospf 100
Device2(config-ospf6)#router-id 2.2.2.2
Device2(config-ospf6)#exit
Device2(config)#interface gigabitethernet1
```

```
Device2(config-if-gigabitethernet1)#ipv6 router ospf 100 area 0
Device2(config-if-gigabitethernet1)#exit
Device2(config)#interface loopback 0
Device2(config-if-loopback0)#ipv6 router ospf 100 area 0
Device2(config-if-loopback0)#exit
```

#Configure Device3.

```
Device3#configure terminal
Device3(config)#ipv6 router ospf 100
Device3(config-ospf6)#router-id 3.3.3.3
Device3(config-ospf6)#exit
Device3(config)#interface gigabitethernet0
Device3(config-if-gigabitethernet0)#ipv6 router ospf 100 area 0
Device3(config-if-gigabitethernet0)#exit
Device3(config)#interface loopback 0
Device3(config-if-loopback0)#ipv6 router ospf 100 area 0
Device3(config-if-loopback0)#exit
```

#View the route table of Device2.

```
Device2#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
     U - Per-user Static route
     O - OSPF, OE-OSPF External, M - Management

L   ::1/128 [0/0]
    via ::, 1w1d:23:51:37, lo0
LC  1::1/128 [0/0]
    via ::, 00:09:34, loopback0
O   2::2/128 [110/2]
    via fe80::201:7aff:fec0:525a, 00:05:29, gigabitethernet1
C   2001:2::/64 [0/0]
    via ::, 00:09:41, gigabitethernet0
L   2001:2::2/128 [0/0]
    via ::, 00:09:39, lo0
C   2001:3::/64 [0/0]
    via ::, 00:08:55, gigabitethernet1
L   2001:3::2/128 [0/0]
    via ::, 00:08:53, lo0
```

#View the route table of Device3.

```
Device3#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
      U - Per-user Static route
      O - OSPF, OE-OSPF External, M - Management

L   ::1/128 [0/0]
    via ::, 1w5d:18:34:53, lo0
O   1::1/128 [110/2]
    via fe80::201:7aff:fe5e:6d2e, 00:29:59, gigabitethernet0
LC  2::2/128 [0/0]
    via ::, 00:32:36, loopback0
C   2001:3::/64 [0/0]
    via ::, 00:32:59, gigabitethernet0
L   2001:3::1/128 [0/0]
    via ::, 00:32:58, lo0
C   2001:4::/64 [0/0]
    via ::, 00:32:44, gigabitethernet1
L   2001:4::1/128 [0/0]
    via ::, 00:32:43, lo0
```

It can be seen that Device2 and Device3 have learned the routE of the peer loopback port by running OSPFv3 protocol, so as to prepare for Device2 and Device3 to establish IBGP neighbors through the loopback port in the next step.

Step 3: Configure the IPv6 BGP basic functions.

#Configure Device1.

Set up the direct-connected EBGP peer with Device2, and introduce 2001:1::/64 to BGP by the network mode.

```
Device1#configure terminal
Device1(config)#router bgp 200
Device1(config-bgp)#bgp router-id 1.1.1.1
Device1(config-bgp)#address-family ipv6
Device1(config-bgp-af)#neighbor 2001:2::2 remote-as 100
Device1(config-bgp-af)#network 2001:1::/64
Device1(config-bgp-af)#exit-address-family
Device1(config-bgp)#exit
```

#Configure Device2.

Establish a direct-connected EBGP peer with Device1, establish a non-direct connected IBGP peer relationship with Device3 through loopback0, and set the next hop of the notification route to itself.

```
Device2(config)#router bgp 100
Device2(config-bgp)#bgp router-id 2.2.2.2
Device2(config-bgp)#address-family ipv6
Device2(config-bgp-af)#neighbor 2001:2::1 remote-as 200
Device2(config-bgp-af)#neighbor 2::2 remote-as 100
Device2(config-bgp-af)#neighbor 2::2 next-hop-self
Device2(config-bgp-af)#exit-address-family
Device2(config-bgp)#neighbor 2::2 update-source loopback 0
Device2(config-bgp)#exit
```

#Configure Device3.

Establish a non-direct IBGP peer relationship with Device2 through loopback0, and introduce 2001:4::/64 into BGP through network.

```
Device3(config)#router bgp 100
Device3(config-bgp)#bgp router-id 3.3.3.3
Device3(config-bgp)#address-family ipv6
Device3(config-bgp-af)#neighbor 1::1 remote-as 100
Device3(config-bgp-af)#network 2001:4::/64
Device3(config-bgp-af)#exit-address-family
Device3(config-bgp)#neighbor 1::1 update-source loopback 0
Device3(config-bgp)#exit
```

#View the IPv6 BGP neighbor status on Device2.

```
Device2#show bgp ipv6 unicast summary
BGP router identifier 2.2.2.2, local AS number 100
BGP table version is 4
2 BGP AS-PATH entries
0 BGP community entries

Neighbor       V    AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down  State/PfxRcd
2::2           4   100       8       6        3    0    0 00:04:12        1
2001:2::1      4   200      15      15        3    0    0 00:11:17        1

Total number of neighbors 2
```

The contents of the State/PfxRcd column are displayed as numbers (the number of routing prefixes received from neighbors). It can be seen that Device2, Device1 and Device3 successfully established IPv6 BGP neighbors

Step 4: Configure the IPSec proposal and the manual tunnel.

#Configure Device1, create IPSec proposal a, adopt ESP transmission encapsulation, encryption algorithm 3DES, authentication algorithm SHA1, create IPSec manual tunnel a, and configure SPI and key.

```
Device1(config)#crypto ipsec proposal a
Device1(config-ipsec-prop)#mode transport
Device1(config-ipsec-prop)#esp 3des sha1
Device1(config-ipsec-prop)#exit
Device1(config)#crypto ipv6-tunnel a manual
Device1(config-manual-tunnel)#set ipsec proposal a
Device1(config-manual-tunnel)#set inbound esp 1000 encryption 0 111111111111111111111111 authentication 0 aaaaaaaaaaaaaaaaaaaa
Device1(config-manual-tunnel)#set outbound esp 1001 encryption 0 aaaaaaaaaaaaaaaaaaaaaaaa authentication 0 11111111111111111111
Device1(config-manual-tunnel)#exit
```

#Configure Device2, create IPSec proposal a, adopt ESP transmission encapsulation, encryption algorithm 3DES, authentication algorithm SHA1, create IPSec manual tunnel a, and configure SPI and key; Create IPSec proposal b, adopt ESP transmission encapsulation, encryption algorithm aes128, authentication algorithm SM3, create IPSec manual tunnel b, and configure SPI and key.

```
Device2(config)#crypto ipsec proposal a
Device2(config-ipsec-prop)#mode transport
Device2(config-ipsec-prop)#esp 3des sha1
Device2(config-ipsec-prop)#exit
Device2(config)#crypto ipv6-tunnel a manual
Device2(config-manual-tunnel)#set ipsec proposal a
Device2(config-manual-tunnel)#set inbound esp 1001 encryption 0 aaaaaaaaaaaaaaaaaaaaaaaa authentication 0 11111111111111111111
Device2(config-manual-tunnel)#set outbound esp 1000 encryption 0 111111111111111111111111 authentication 0 aaaaaaaaaaaaaaaaaaaa
Device2(config-manual-tunnel)#exit
Device2(config)#crypto ipsec proposal b
Device2(config-ipsec-prop)#mode transport
Device2(config-ipsec-prop)#esp aes128 sm3
Device2(config-ipsec-prop)#exit
Device2(config)#crypto ipv6-tunnel b manual
Device2(config-manual-tunnel)#set ipsec proposal b
```

```
Device2(config-manual-tunnel)#set inbound esp 2001 encryption 0 1111111111111111 authentication 0 aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa

Device2(config-manual-tunnel)#set outbound esp 2000 encryption 0 1111111111111111 authentication 0 11111111111111111111111111111111

Device2(config-manual-tunnel)#exit
```

#Configure Device3, create IPSec proposal b, adopt ESP transmission encapsulation, encryption algorithm aes128, authentication algorithm sm3, create IPSec manual tunnel b, and configure SPI and key.

```
Device3(config)#crypto ipsec proposal b

Device3(config-ipsec-prop)#mode transport

Device3(config-ipsec-prop)#esp aes128 sm3

Device3(config-ipsec-prop)#exit

Device3(config)#crypto ipv6-tunnel b manual

Device3(config-manual-tunnel)#set ipsec proposal b

Device3(config-manual-tunnel)#set inbound esp 2000 encryption 0 1111111111111111 authentication 0 11111111111111111111111111111111

Device3(config-manual-tunnel)#set outbound esp 2001 encryption 0 1111111111111111 authentication 0 aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa

Device3(config-manual-tunnel)#exit
```

Step 5: Configure the BGP neighbor to bind the corresponding the IPSec tunnel.

#Configure Device1.

Configure the neighbor 2001:2::2 to enable the ipsec-tunnel protection communication.

```
Device1#configure terminal

Device1(config)#router bgp 200

Device1(config-bgp)#address-family ipv6

Device1(config-bgp-af)#neighbor 2001:2::2 ipsec-tunnel a

Device1(config-bgp-af)#exit-address-family

Device1(config-bgp)#exit
```

#Configure Device2.

Configure the neighbors 2001:2::2 and 2::2 to enable the corresponding ipsec-tunnel protection communication respectively.

```
Device2(config)#router bgp 100

Device2(config-bgp)#bgp router-id 2.2.2.2

Device2(config-bgp)#address-family ipv6

Device2(config-bgp-af)#neighbor 2001:2::1 ipsec-tunnel a

Device2(config-bgp-af)#neighbor 2::2 ipsec-tunnel b

Device2(config-bgp-af)#exit-address-family

Device2(config-bgp)#exit
```

#Configure Device3.

Configure the neighbor 1::1 to enable the ipsec-tunnel protection communication.

```
Device3(config)#router bgp 100
Device3(config-bgp)#address-family ipv6
Device3(config-bgp-af)#neighbor 1::1 ipsec-tunnel b
Device3(config-bgp-af)#exit-address-family
Device3(config-bgp)#exit
```

Step 6: Check the result.

#View the IPv6 BGP process neighbor information of Device2.

```
Device2#show bgp ipv6 unicast summary
BGP router identifier 2.2.2.2, local AS number 100
BGP table version is 9
2 BGP AS-PATH entries
0 BGP community entries

Neighbor      V       AS MsgRcvd MsgSent  TblVer  InQ OutQ Up/Down  State/PfxRcd
2::2          4       100      3       3       9    0    0 00:01:32       1
2001:2::1     4       200     22      22       9    0    0 00:17:04       1

Total number of neighbors 2
```

You can see that the neighbors with Device1 and Device3 are all up.

#View the IPv6 BGP route learned by Device1.

```
Device1#show ipv6 route bgp
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management

B   2001:4::/64 [20/0]
      via 2001:2::2, 00:00:02, gigabitethernet1
```

You can see that the interface direct-connected route 2001:4::/64 to Device3 is learned.

#View the IPv6 BGP route learned by Device3.

```
Device1#show ipv6 route bgp
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management
```

```
B   2001:1::/64 [200/0]
    via 1::1, 00:00:28, gigabitethernet0
```

You can see that the interface direct-connected route 2001:1::/64 to Device1 is learned.

12.3.10. Configure IPv6 BGP Static Fast Re-routing

Network Requirements

- All devices are configured with the IPv6 BGP protocol.
- Enable the static fast rerouting between Device1 and Device3. When the line fails, the service can be quickly switched to Device2 for communication.

Network Topology

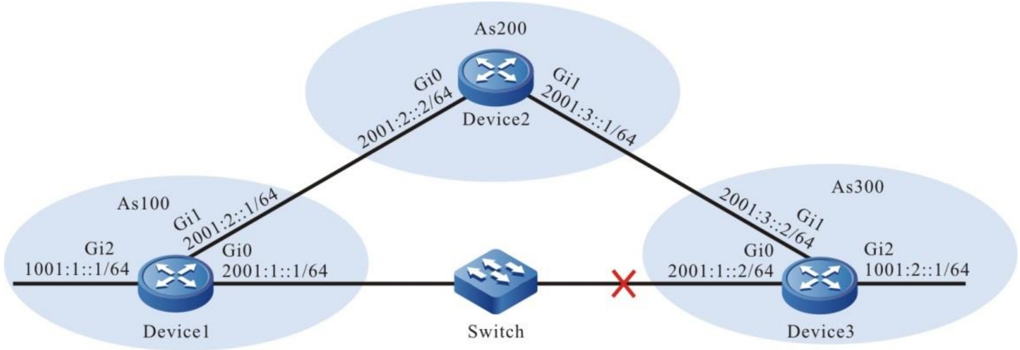

Figure 12-10 Networking of configuring IPv6 BGP static fast re-routing

Configuration Steps

Step 1: Configure the IPv6 address of the interface (omitted).

Step 2: Configure BGP.

#Configure Device1 to set up the BGP neighbor with Device2 and Device3.

```
Device1(config)#router bgp 100
Device1(config-bgp)#bgp router-id 1.1.1.1
Device1(config-bgp)#address-family ipv6
Device1(config-bgp-af)#neighbor 2001:1::2 remote-as 300
Device1(config-bgp-af)#neighbor 2001:2::2 remote-as 200
Device1(config-bgp-af)#exit-address-family
Device1(config-bgp)#exit
```

#Configure Device2 to set up the BGP neighbor with Device1 and Device3.

```
Device2(config)#router bgp 200
Device2(config-bgp)#bgp router-id 2.2.2.2
Device2(config-bgp)#address-family ipv6
Device2(config-bgp-af)#neighbor 2001:2::1 remote-as 100
Device2(config-bgp-af)#neighbor 2001:3::2 remote-as 300
Device2(config-bgp-af)#exit-address-family
Device2 (config-bgp)#exit
```

#Configure Device3 to set up the BGP neighbor with Device1 and Device2.

```
Device3(config)#router bgp 300
Device3(config-bgp)#bgp router-id 3.3.3.3
Device3(config-bgp)#address-family ipv6 unicast
Device3(config-bgp-af)#neighbor 2001:3::1 remote-as 200
Device3(config-bgp-af)#neighbor 2001:1::1 remote-as 100
Device3(config-bgp-af)#network 1001:2::/64
Device3(config-bgp-af)#exit-address-family
Device3(config-bgp)#exit
```

Step 3: On the interface gigabitethernet0 of Device3, configure the echo function of ipv6 bfd.

```
Device3(config)#interface gigabitethernet0
Device3(config-if-gigabitethernet0)# ipv6 bfd echo
Device3(config-if-gigabitethernet0)#exit
```

Step 4: Configure the routing policy.

#Configure Device1, configure route-map, and call ACL to match only 1001:2::1/64, while other network segments will be filtered out. The routing application matching the match rule backs up the next-hop interface gigabitethernet1, and the next-hop address 2001:2::2.

```
Device1(config)#ipv6 access-list extended 7001
Device1(config-v6-list)#permit ipv6 1001:2::1/64 any
Device1(config-v6-list)#exit
Device1(config)#route-map ipv6frr_bgp
Device1(config-route-map)#match ipv6 address 7001
Device1(config-route-map)#set ipv6 fast-reroute backup-interface gigabitethernet1
backup-nexthop 2001:2::2
Device1(config-route-map)#exit
```

Step 5: Configure the static fast re-routing.

```
Device1(config)#router bgp 100
Device1(config-bgp)#address-family ipv6 unicast
Device1(config-bgp-af)#fast-reroute route-map ipv6frr_bgp
Device1(config-bgp-af)#exit-address-family
Device1(config-bgp)#exit
```

Step 6: Check the result.

#View the IPv6 BGP route table of Device1.

```
Device1#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
    U - Per-user Static route
```

```
          O - OSPF, OE-OSPF External, M - Management
L    ::1/128 [0/0]
     via ::, 23:12:10, lo0
C    1001:1::/64 [0/0]
     via ::, 23:07:17, gigabitethernet2
L    1001:1::1/128 [0/0]
     via ::, 23:07:17, gigabitethernet2
B    1001:2::/64 [20/0]
     via 2001:1::2, 00:03:25, gigabitethernet0
C    2001:1::/64 [0/0]
     via ::, 13:47:06, gigabitethernet0
L    2001:1::1/128 [0/0]
     via ::, 13:47:06, gigabitethernet0
C    2001:2::/64 [0/0]
     via ::, 23:06:35, gigabitethernet1
L    2001:2::1/128 [0/0]
     via ::, 23:06:35, gigabitethernet1
```

It can be seen from the routing table that route 1001:2::/64 preferably communicates with the line between Device1 and Device3.

#View the IPv6 FRR table of Device1.

```
Device1#show ipv6 frr route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
          U - Per-user Static route
          O - OSPF, OE-OSPF External, M - Management
B    1001:2::/64 [20/4294967295]
     via 2001:2::2, 00:05:16, gigabitethernet1
```

You can see that the next-hop of the frr route is 2001:2::2, and the outgoing interface is gigabitethernet1.

#View the BFD session information of Device1.

```
Device1#show bfd session ipv6 detail
Total ipv6 session number: 1
Device1#show bfd session ipv6
OurAddr                    NeighAddr         LD/RD           State      Holddown
Interface
2001:1::1                      2001:1::2          1027/1027          UP          500
gigabitethernet0
Type:ipv6 direct  Mode:echo
Local Discriminator:73  Remote Discriminator:73
Local State:UP  Remote State:UP  Up for: 0h:8m:52s  Number of times UP:1
Send Interval:100ms  Detection time:500ms(100ms*5)
Local Diag:0  Demand mode:0  Poll bit:0
Registered protocols:FIB_MGR
Agent session info:
 Sender:slot 0  Recver:slot 0
```

You can see that FIB_MGR is linked with BFD successfully, the session is set up normally, and the session mode is echo.

#After the line between Device1 and Device3 fails, it can quickly detect and switch to Device2 for communication. Check the routing table of Device1. In the routing table, the outgoing interface to the destination network 1001:2::/64 has been switched to the backup interface gigabitethernet1.

```
Device1#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management
L   ::1/128 [0/0]
    via ::, 23:18:47, lo0
C   1001:1::/64 [0/0]
    via ::, 23:13:54, gigabitethernet2
L   1001:1::1/128 [0/0]
    via ::, 23:13:54, gigabitethernet2
B   1001:2::/64 [20/0]
    via 2001:2::2, 00:00:03, gigabitethernet1
C   2001:2::/64 [0/0]
    via ::, 23:13:12, gigabitethernet1
L   2001:2::1/128 [0/0]
    via ::, 23:13:12, gigabitethernet1
```

12.3.11. Configure IPv6 BGP Dynamic Fast Re-routing

Network Requirements

- All devices are configured with the IPv6 BGP protocol.
- Enable the dynamic fast rerouting between Device1 and Device3. When the line fails, the service can be quickly switched to Device2 for communication.

Network Topology

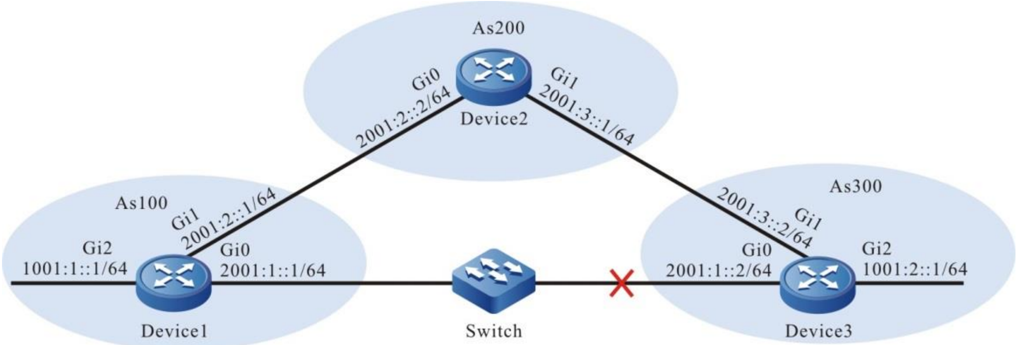

Figure 12-11Networking of configuring the IPv6 BGP dynamic fast re-routing

Configuration Steps

Step 1: Configure the IPv6 address of the interface (omitted).

Step 2: Configure BGP.

#Configure Device1 to set up the BGP neighbor with Device2 and Device3.

```
Device1(config)#router bgp 100
Device1(config-bgp)#bgp router-id 1.1.1.1
Device1(config-bgp)#address-family ipv6
Device1(config-bgp-af)#neighbor 2001:1::2 remote-as 300
Device1(config-bgp-af)#neighbor 2001:2::2 remote-as 200
Device1(config-bgp-af)#exit-address-family
Device1(config-bgp)#exit
```

#Configure Device2 to set up the BGP neighbor with Device1 and Device3.

```
Device2(config)#router bgp 200
Device2(config-bgp)#bgp router-id 2.2.2.2
Device2(config-bgp)#address-family ipv6
Device2(config-bgp-af)#neighbor 2001:2::1 remote-as 100
Device2(config-bgp-af)#neighbor 2001:3::2 remote-as 300
Device2(config-bgp-af)#exit-address-family
Device2 (config-bgp)#exit
```

#Configure Device3 to set up the BGP neighbor with Device1 and Device2.

```
Device3(config)#router bgp 300
Device3(config-bgp)#bgp router-id 3.3.3.3
Device3(config-bgp)#address-family ipv6 unicast
Device3(config-bgp-af)#neighbor 2001:3::1 remote-as 200
Device3(config-bgp-af)#neighbor 2001:1::1 remote-as 100
Device3(config-bgp-af)#network 1001:2::/64
Device3(config-bgp-af)#exit-address-family
```

```
Device3(config-bgp)#exit
```

Step 3: On the interface gigabitethernet0 of Device3, configure the echo function of ipv6 bfd.

```
Device3(config)#interface gigabitethernet0
Device3(config-if-gigabitethernet0)# ipv6 bfd echo
Device3(config-if-gigabitethernet0)#exit
```

Step 4: Configure the routing policy.

#Configure Device1, configure route-map, and call ACL to match only 1001:2::1/64, while other network segments will be filtered out. The routing application matching the match rule backs up the next-hop interface gigabitethernet1, and the next-hop address 2001:2::2.

```
Device1(config)#ipv6 access-list extended 7001
Device1(config-v6-list)#permit ipv6 1001:2::1/64 any
Device1(config-v6-list)#exit
Device1(config)#route-map ipv6frr_bgp
Device1(config-route-map)#match ipv6 address 7001
Device1(config-route-map)# set ipv6 fast-reroute backup-nexthop auto
Device1(config-route-map)#exit
```

Step 5: Configure the static fast re-routing.

```
Device1(config)#router bgp 100
Device1(config-bgp)#address-family ipv6 unicast
Device1(config-bgp-af)#fast-reroute route-map ipv6frr_bgp
Device1(config-bgp-af)#exit-address-family
Device1(config-bgp)#exit
```

Step 6: Check the result.

#View the IPv6 BGP route table of Device1.

```
Device1#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
     U - Per-user Static route
     O - OSPF, OE-OSPF External, M - Management
L   ::1/128 [0/0]
    via ::, 23:12:10, lo0
C   1001:1::/64 [0/0]
    via ::, 23:07:17, gigabitethernet2
L   1001:1::1/128 [0/0]
    via ::, 23:07:17, gigabitethernet2
B   1001:2::/64 [20/0]
    via 2001:1::2, 00:03:25, gigabitethernet0
```

```
C   2001:1::/64 [0/0]
    via ::, 13:47:06, gigabitethernet0
L   2001:1::1/128 [0/0]
    via ::, 13:47:06, gigabitethernet0
C   2001:2::/64 [0/0]
    via ::, 23:06:35, gigabitethernet1
L   2001:2::1/128 [0/0]
    via ::, 23:06:35, gigabitethernet1
```

It can be seen from the routing table that route 1001:2::/64 preferably communicates with the line between Device1 and Device3.

#View the IPv6 FRR table of Device1.

```
Device1#show ipv6 frr route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management
B   1001:2::/64 [20/4294967295]
    via 2001:2::2, 00:05:16, gigabitethernet1
```

You can see that the next-hop of the frr route is 2001:2::2, and the outgoing interface is gigabitethernet1.

#View the BFD session information of Device1.

```
Device1#show bfd session ipv6 detail
Total ipv6 session number: 1
Device1#show bfd session ipv6
OurAddr                   NeighAddr       LD/RD          State    Holddown
Interface
2001:1::1                    2001:1::2       1023/1023        UP        500
gigabitethernet0
Type:ipv6 direct  Mode:echo
Local Discriminator:73  Remote Discriminator:73
Local State:UP  Remote State:UP  Up for: 0h:8m:52s  Number of times UP:1
Send Interval:100ms  Detection time:500ms(100ms*5)
Local Diag:0  Demand mode:0  Poll bit:0
Registered protocols:FIB_MGR
Agent session info:
 Sender:slot 0  Recver:slot 0
```

You can see that FIB_MGR is linked with BFD successfully, the session is set up normally, and the session mode is echo.

#After the line between Device1 and Device3 fails, it can quickly detect and switch to Device2 for communication. Check the routing table of Device1. In the routing table, the outgoing interface to the destination network 1001:2::/64 has been switched to the backup interface gigabitethernet1.

```
Device1#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
     U - Per-user Static route
     O - OSPF, OE-OSPF External, M - Management
L   ::1/128 [0/0]
    via ::, 23:18:47, lo0
C   1001:1::/64 [0/0]
    via ::, 23:13:54, gigabitethernet2
L   1001:1::1/128 [0/0]
    via ::, 23:13:54, gigabitethernet2
B   1001:2::/64 [20/0]
    via 2001:2::2, 00:00:03, gigabitethernet1
C   2001:2::/64 [0/0]
    via ::, 23:13:12, gigabitethernet1
L   2001:2::1/128 [0/0]
    via ::, 23:13:12, gigabitethernet1
```

13. PBR

13.1. Overview

PBR (policy-based-route) is one technology of forwarding packets according to the configured packet processing policy (including match policy and forwarding policy). When forwarding packets, first match the packet according to the match policy. If matching successfully, forward the packet according to the forwarding policy. The match policy is based on the standard and extended ACL. Forwarding policy includes setting the priority of the IP packet, setting the sending interface of the packet, setting the forwarding next hop of the packet, setting the forwarding of the packet in the specified VPN instance, and so on.

The PBR is divided to forwarding PBR and local PBR:

- Forwarding PBR: It takes effect only for the packet received by the interface that applies the PBR, but does not take effect for the local packet.
- Local PBR: It takes effect only for the local packet, but does not take effect for the forwarded packet.

The priority of the PBR is higher than the common route, that is, the packet first executes the PBR processing. If the packet does not match the PBR match policy, forward according to the common route.

13.2. PBR Function Configuration

Table 13-1 PBR function list

Configuration Task	
Configure the PBR	Create the PBR policy and rule
	Configure the match policy
	Configure the forwarding policy
Apply the PBR	Apply the forwarding PBR
	Apply the local PBR

13.2.1. Configure PBR

Configuration Condition

None

Create PBR Policy and Rule

When creating the PBR rule, you should specify the serial number, the smaller the serial number, the higher the rule priority.

Table 13-2 Create the PBR policy and rule

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Create the PBR policy and rule	**route-policy** *policy-name* **permit** *sequence*	Mandatory By default, do not create the PBR policy and rule.

Configure Match Policy

The packet executes the PBR operation according to the priority of the PBR rule from high to low. The PBR rules include match policy and forwarding policy. If the packet matches successfully, forward the packet according to the forwarding policy, but do not execute the next PBR rule. If the PBR rule does not configure the forwarding policy, execute the next rule no matter whether the packet matches.

Table 13-3 Configure the match policy

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the PBR configuration mode	**route-policy** *policy-name* **permit** *sequence*	-
Configure the match policy	**match ip address acl** { *access-list-number* | *access-list-name* }	Optional By default, do not configure the match policy.

Note:

- The match policy of each rule can only associate one ACL.
- When configuring the match policy and if the specified ACL does not exist, the command can be configured successfully, but when all packets fail to match, the match policy does not take effect. The match policy takes effect only after the specified ACL is created.

Configure Forwarding Policy

The device supports five forwarding policies, that is, set the packet to be forwarded in the specified VPN instance, set the priority of the IP packet, set the sending next-hop address of the packet, set the sending interface of the packet, set the load balance mode of the packet, drop packet.

- Set the packet to be forwarded in the specified VPN instance: The matched packet searches for the forwarding information in the specified VPN instance according to the destination address of the packet. If finding the forwarding information, forward the packet according to the forwarding information. Otherwise, drop the packet.
- Set the priority of the IP packet: Modify the priority of the IP packet to the specified value
- Set the sending next-hop address of the packet: Send the packet to the specified next hop
- Set the sending interface of the packet: Send the packet from the specified interface
- Set the load balance mode of the packet: Adopt the specified load balance mode to forward the packet
- Drop the packet: Directly drop the packet, but do not perform other processing.

Table 13-4 Configure the forwarding policy

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the PBR configuration mode	**route-policy** *policy-name* **permit** *sequence*	-
Set the packet to be forwarded in the specified VPN instance	**set ip vrf** { *vrf-name-list* }	Optional By default, do not set the packet to be forwarded in the specified VPN instance.
Set the priority of the IP packet	**set ip precedence** { *precedence-number* | *precedence-type* }	Optional By default, do not set the priority of the IP packet.
Set the sending next-hop address of the packet	**set ip next-hop** { *ip-address-list* } [**track** { *track-id* }]	Optional By default, do not set the sending next-hop address of the packet.
Set the sending interface of the packet	**set interface** { *interface-name-list* } [**track** { *track-id* }]	Optional By default, do not set the sending interface of the packet.

Step	Command	Description
Set the load balance mode of the packet	**set load-sharing** { per-packet | per-source }	Optional By default, do not adopt any load balance mode. Select the first valid forwarding path to forward the packet.
Drop the packet	**set drop**	Optional By default, do not set dropping the packet.

Note:

- One rule can specify multiple VPN instances. When processing the packet according to the forwarding policy, search according to the configured order.
- If setting the packet to be forwarded in the specified VPN instance, do not execute other forwarding policty except for setting the packet priority.
- The order of executing the operations of setting the sending next-hop address of the packet, setting the sending interface of the packet, and dropping the packet: set the sending next-hop address of the packet > set the sending interface of the packet > drop the packet
- When setting the sending next-hop address of the packet, you can specify one or multiple next hops and the direct-connected next hop is the valid next hop. When specifying multiple next hops, search for the valid next hop according to the configuration order. After finding the valid next hop, send the packet to the next hop.
- When setting the sending interface of the packet, you can specify one or multiple interfaces. The UP interface is the valid sending interface. When specifying multiple sending interfaces, search for the valid sending interface according to the configuration order. After finding the valid sending interface, send the packet from the interface. If the sending interface is Ethernet, only the packet whose destination address is the direct-connected next-hop address of the sending interface can be forwarded successfully. Otherwise, drop the packet.
- Set the load balance mode of the packet: The set sending next hop of the packet and the set sending interface of the packet calculate the load forwarding path respectively, but do not take the sending next hop of the packet and the sending interface of the packet as the load path with the same priority to calculate. Send the packet according to the following order: Set the sending next-hop address of the packet > Set the sending interface of the packet > drop the packet.
- When setting the next hop address or sending interface of the packet to link with the track, the set next hop address or sending interface may become valid only when the track status becomes up.
- The process of setting bound and unbound tracks does not support updating. For example, if **set ip next-hop 1.1.1.1** is set first, you need to cancel the setting before successfully setting **set ip next-hop 1.1.1.1 track 1**.

13.2.2. Apply PBR

Configuration Condition

None

Apply Forwarding PBR

Apply the forwarding PBR on the specified interface. The packet received from the interface is forwarded according to the specified PBR policy.

Table 13-5 Apply the forwarding PBR

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the interface configuration mode.	**interface** *interface-name*	-
Apply the forwarding PBR	**ip policy** *policy-name* [**any**]	Mandatory By default, the interface does not apply the forwarding PBR.

Note:

- If the applied PBR policy of the interface does not exist, the command can be configured successfully, but the packet is forwarded according to the common route. After the applied PBR policy of the interface is created, the packet is forwarded according to the PBR policy.
- If not configuring any, the interface receives the local packet, but does not execute the PBR processing. If configuring any, the interface receives the local packet and executes the PBR processing.

Apply Local PBR

The local PBR only processes the packet generated by the device.

Table 13-6 Apply the local PBR

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Apply the local PBR	**ip local policy** *policy-name*	Mandatory By default, do not apply the local PBR.

Note:

- If the applied PBR policy of the interface does not exist, the command can be configured successfully, but the packet generated by the device is forwarded according to the common route. After the applied PBR policy is created, the packet is forwarded according to the PBR policy.

13.2.3. PBR Monitoring and Maintaining

Table 13-7 PBR monitoring and maintaining

Command	Description
show ip local policy	Display the configuration information of the local PBR
show ip policy [*interface-name*]	Display the configuration information of the applied forwarding PBR
show route-policy [*policy-name*]	Display the configuration information of the PBR policy and rule

13.3. PBR Typical Configuration Example

13.3.1. Configure Local PBR

Network Requirements

- Run the OSPF protocol on all devices and configure the local PBR on Device1.
- There is the server with the IP address 2.2.2.2/24 in the IP network.
- Configure the local PBR so that Device1 can access the server 2.2.2.2 via Device2.

Network Topology

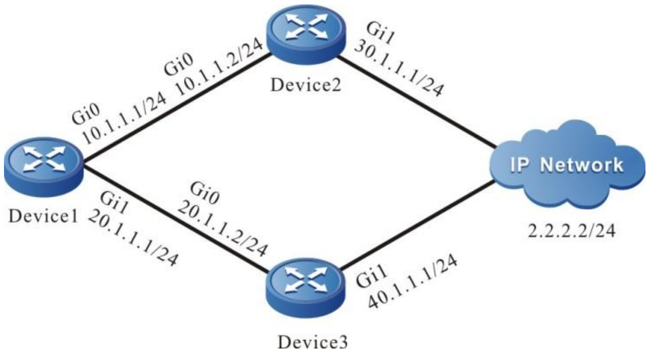

Figure 13-1 Networking for configuring the local PBR

Configuration Steps

Step 1: Configure the IP address of the interface (omitted).

Step 2: Enable the unitcast route protocol OSPF so that all devices in the network can communicate with each other.

#Configure Device1.

```
Device1#configure terminal
Device1(config)#router ospf 100
Device1(config-ospf)#network 10.1.1.0 0.0.0.255 area 0
Device1(config-ospf)#network 20.1.1.0 0.0.0.255 area 0
Device1(config-ospf)#exit
```

#Configure Device2.

```
Device2#configure terminal
Device2(config)#router ospf 100
Device2(config-ospf)#network 10.1.1.0 0.0.0.255 area 0
Device2(config-ospf)#network 30.1.1.0 0.0.0.255 area 0
Device2(config-ospf)#exit
```

#Configure Device3.

```
Device3#configure terminal
Device3(config)#router ospf 100
Device3(config-ospf)#network 20.1.1.0 0.0.0.255 area 0
Device3(config-ospf)#network 40.1.1.0 0.0.0.255 area 0
Device3(config-ospf)#exit
```

#View the route table of Device1 and you can see that there are two next hops to the network 2.2.2.0/24.

```
Device1#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

O   2.2.2.0/24 [110/3] via 10.1.1.2, 00:00:09, gigabitethernet0
             [110/3] via 20.1.1.2, 00:00:09, gigabitethernet1
C   10.1.1.0/24 is directly connected, 21:41:21, gigabitethernet0
L   10.1.1.1/32 is directly connected, 21:41:21, gigabitethernet0
C   20.1.1.0/24 is directly connected, 15:19:15, gigabitethernet1
L   20.1.1.1/32 is directly connected, 15:19:15, gigabitethernet1
O   30.1.1.0/24 [110/2] via 10.1.1.2, 18:55:36, gigabitethernet0
O   40.1.1.0/24 [110/2] via 20.1.1.2, 00:22:08, gigabitethernet1
C   127.0.0.0/8 is directly connected, 87:42:47, lo0
L   127.0.0.1/32 is directly connected, 87:42:47, lo0
```

#Configure Device1, and modify the cost value of the interface gigabitethernet0 to 100 to make the route to the network 2.2.2.0/24 first select the interface gigabitethernet1.

```
Device1(config)#interface gigabitethernet0
Device1(config-if-gigabitethernet0)#ip ospf cost 100
Device1(config-if-gigabitethernet0)#exit
```

#View the route table of Device1.

```
Device1#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

O   2.2.2.0/24 [110/3] via 20.1.1.2, 01:12:50, gigabitethernet1
C   10.1.1.0/24 is directly connected, 22:54:03, gigabitethernet0
L   10.1.1.1/32 is directly connected, 22:54:03, gigabitethernet0
C   20.1.1.0/24 is directly connected, 16:31:57, gigabitethernet1
L   20.1.1.1/32 is directly connected, 16:31:57, gigabitethernet1
O   30.1.1.0/24 [110/3] via 20.1.1.2, 00:31:42, gigabitethernet0
O   40.1.1.0/24 [110/2] via 20.1.1.2, 01:34:50, gigabitethernet1
C   127.0.0.0/8 is directly connected, 88:55:28, lo0
L   127.0.0.1/32 is directly connected, 88:55:28, lo0
```

#On Device1, use the Traceroute command to view the path to the server 2.2.2.2.

```
Device1#traceroute 2.2.2.2
Type escape sequence to abort.
Tracing the route to 2.2.2.2 , min ttl = 1, max ttl = 30 .

 1  20.1.1.2   0 ms    0 ms    16 ms
……
n  2.2.2.2    0 ms    0 ms    0 ms
```

You can see that Device1 access the server 2.2.2.2 via Device3.

Step 3: Configure the PBR on Device1.

#Configure ACL 1001, permitting the device to access the network 2.2.2.0/24.

```
Device1(config)#ip access-list extended 1001
Device1(config-ext-nacl)#permit ip any 2.2.2.0 0.0.0.255
Device1(config-ext-nacl)#exit
```

#Configure PBR aaa, associate with the ACL 1001, and specify the next hop as 10.1.1.2.

```
Device1(config)#route-policy aaa permit 10
Device1(config-pbr)#match ip address acl 1001
Device1(config-pbr)#set ip next-hop 10.1.1.2
```

```
Device1(config-pbr)#exit
```

#View the information of Device1 PBR aaa.

```
Device1#show route-policy aaa
route-policy aaa
  sequence 10 permit:
    match ip address acl 1001
    set ip next-hop 10.1.1.2
```

Step 4: Apply the PBR.

#Apply the local PBR aaa on Device1.

```
Device1(config)#ip local policy aaa
```

Step 5: Check the result.

#On Device1, use the Traceroute command to view the path to the server 2.2.2.2.

```
Device1#traceroute 2.2.2.2
Type escape sequence to abort.
Tracing the route to 2.2.2.2 , min ttl = 1, max ttl = 30 .

 1  10.1.1.2   0 ms    0 ms    0 ms
……
 n  2.2.2.2    0 ms    0 ms    0 ms
```

You can see that after applying the local PBR, Device1 accesses the server 2.2.2.2 via Device2.

13.3.2. Configure Forwarding PBR

Network Requirements

- Run the OSPF protocol on all devices and configure the forwarding PBR on Device1.
- There is the server with the IP address 2.2.2.2/24 in the IP network.
- Configure the forwarding PBR so that PC can access the server 2.2.2.2 via Device1, Device2.

Network Topology

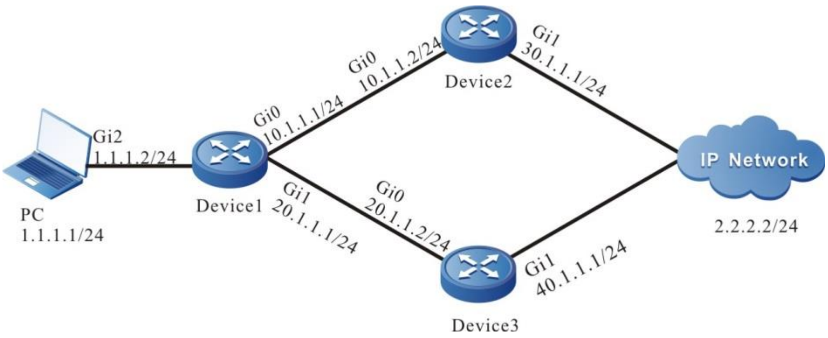

Figure 13–1 Networking for configuring the forwarding PBR

Configuration Steps

Step 1: Configure the IP address of the interface (omitted).

Step 2: Enable the unitcast route protocol OSPF so that all devices in the network can communicate with each other.

#Configure Device1.

```
Device1#configure terminal
Device1(config)#router ospf 100
Device1(config-ospf)#network 1.1.1.0 0.0.0.255 area 0
Device1(config-ospf)#network 10.1.1.0 0.0.0.255 area 0
Device1(config-ospf)#network 20.1.1.0 0.0.0.255 area 0
Device1(config-ospf)#exit
```

#Configure Device2.

```
Device2#configure terminal
Device2(config)#router ospf 100
Device2(config-ospf)#network 10.1.1.0 0.0.0.255 area 0
Device2(config-ospf)#network 30.1.1.0 0.0.0.255 area 0
Device2(config-ospf)#exit
```

#Configure Device3.

```
Device3#configure terminal
Device3(config)#router ospf 100
Device3(config-ospf)#network 20.1.1.0 0.0.0.255 area 0
Device3(config-ospf)#network 40.1.1.0 0.0.0.255 area 0
Device3(config-ospf)#exit
```

#View the route table of Device1 and you can see that there are two next hops to the network 2.2.2.0/24.

```
Device1#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external


C   1.1.1.0/24 is directly connected, 22:14:53, gigabitethernet2
L   1.1.1.1/32 is directly connected, 22:14:53, gigabitethernet2
O   2.2.2.0/24 [110/3] via 10.1.1.2, 00:00:09, gigabitethernet0
               [110/3] via 20.1.1.2, 00:00:09, gigabitethernet1
C   10.1.1.0/24 is directly connected, 21:41:21, gigabitethernet0
L   10.1.1.1/32 is directly connected, 21:41:21, gigabitethernet0
C   20.1.1.0/24 is directly connected, 15:19:15, gigabitethernet1
```

```
L   20.1.1.1/32 is directly connected, 15:19:15, gigabitethernet1
O   30.1.1.0/24 [110/2] via 10.1.1.2, 18:55:36, gigabitethernet0
O   40.1.1.0/24 [110/2] via 20.1.1.2, 00:22:08, gigabitethernet1
C   127.0.0.0/8 is directly connected, 87:42:47, lo0
L   127.0.0.1/32 is directly connected, 87:42:47, lo0
```

#Configure Device1, and modify the cost value of the interface gigabitethernet0 to 100 to make the route to the network 2.2.2.0/24 first select the interface gigabitethernet1.

```
Device1(config)#interface gigabitethernet0
Device1(config-if-gigabitethernet0)#ip ospf cost 100
Device1(config-if-gigabitethernet0)#exit
```

#View the route table of Device1.

```
Device1#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

C   1.1.1.0/24 is directly connected, 23:27:34, gigabitethernet2
L   1.1.1.1/32 is directly connected, 23:27:34, gigabitethernet2
O   2.2.2.0/24 [110/3] via 20.1.1.2, 01:12:50, gigabitethernet1
C   10.1.1.0/24 is directly connected, 22:54:03, gigabitethernet0
L   10.1.1.1/32 is directly connected, 22:54:03, gigabitethernet0
C   20.1.1.0/24 is directly connected, 16:31:57, gigabitethernet1
L   20.1.1.1/32 is directly connected, 16:31:57, gigabitethernet1
O   30.1.1.0/24 [110/3] via 20.1.1.2, 00:31:42, gigabitethernet0
O   40.1.1.0/24 [110/2] via 20.1.1.2, 01:34:50, gigabitethernet1
C   127.0.0.0/8 is directly connected, 88:55:28, lo0
L   127.0.0.1/32 is directly connected, 88:55:28, lo0
```

#On PC, use the Traceroute command to view the path to the server 2.2.2.2.

```
C:\Documents and Settings\Administrator>tracert 2.2.2.2

Tracing route to 2.2.2.2 over a maximum of 30 hops

 1    1 ms     1 ms     1 ms  1.1.1.2
 2   <1 ms    <1 ms    <1 ms  20.1.1.2
 ……
 n   <1 ms    <1 ms    <1 ms  2.2.2.2
Trace complete.
```

You can see that PC accesses the server 2.2.2.2 via Device1 and Device3.

Step 3: Configure the PBR on Device1.

#Configure the ACL 1001, permitting the PC to access the network 2.2.2.0/24.

```
Device1(config)#ip access-list extended 1001
Device1(config-ext-nacl)#permit ip any 2.2.2.0 0.0.0.255
Device1(config-ext-nacl)#exit
```

#Configure PBR aaa, associate with the ACL 1001, and specify the next hop as 10.1.1.2.

```
Device1(config)#route-policy aaa permit 10
Device1(config-pbr)#match ip address acl 1001
Device1(config-pbr)#set ip next-hop 10.1.1.2
Device1(config-pbr)#exit
```

#View the information about the PBR aaa of Device1.

```
Device1#show route-policy aaa
route-policy aaa
  sequence 10 permit:
    match ip address acl 1001
    set ip next-hop 10.1.1.2
```

Step 4: Apply the PBR.

#Apply the PBR aaa to the interface gigabitethernet2 of Device1.

```
Device1(config)#interface gigabitethernet2
Device1(config-if-gigabitethernet2)#ip policy aaa
Device1(config-if-gigabitethernet2)#exit
```

Step 5: Check the result.

#On PC, use the Traceroute command to view the path to the server 2.2.2.2.

```
C:\Documents and Settings\Administrator>tracert 2.2.2.2


Tracing route to 2.2.2.2 over a maximum of 30 hops


  1    1 ms     1 ms     1 ms  1.1.1.2
  2   <1 ms    <1 ms    <1 ms  10.1.1.2
  ......
  n   <1 ms    <1 ms    <1 ms  2.2.2.2
Trace complete.
```

You can see that after applying the PBR to the interface gigabitethernet2, the PC accesses the server 2.2.2.2 via Device1, Device2.

Note:

- The forwarding PBR needs to be configured on the interface receiving the packet.

13.3.3. Configure Forwarding Policy Route to Link with TRACK

Network Requirements

- The OSPF protocol is running on all devices, forwarding policy routing is configured on Device1, and there are servers with IP address 2.2.2.2/24 in IP network.
- Device1 simultaneously monitors the interface bandwidth utilization of BSM and gigabitethernet2 and the BFD session status between Device1 and Device2 through track. When the link between Device1 and Device2 is normal, the PC accesses server 2.2.2.2 through Device1 and Device2. When the link between Device1 and Device2 fails, the PC accesses server 2.2.2.2 through Device1 and Device3.

Network Topology

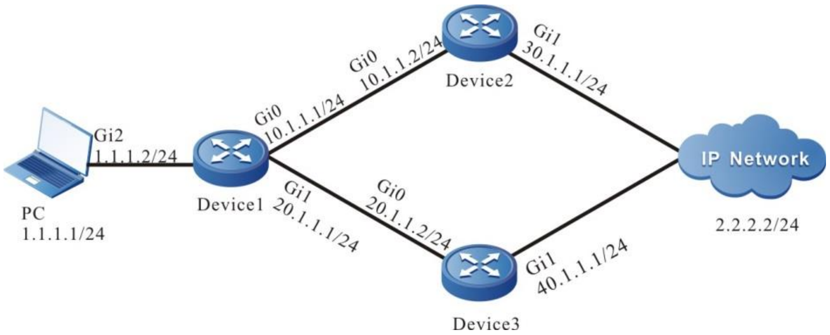

Figure 13-2 Networking of configuring the forwarding policy route to link with TRACK

Configuration Steps

Step 1: Configure the IP address of the interface (omitted).

Step 2: Enable the unitcast route protocol OSPF so that all devices in the network can communicate with each other.

#Configure Device1.

```
Device1#configure terminal
Device1(config)#router ospf 100
Device1(config-ospf)#network 1.1.1.0 0.0.0.255 area 0
Device1(config-ospf)#network 10.1.1.0 0.0.0.255 area 0
Device1(config-ospf)#network 20.1.1.0 0.0.0.255 area 0
Device1(config-ospf)#exit
```

#Configure Device2.

```
Device2#configure terminal
Device2(config)#router ospf 100
Device2(config-ospf)#network 10.1.1.0 0.0.0.255 area 0
Device2(config-ospf)#network 30.1.1.0 0.0.0.255 area 0
Device2(config-ospf)#exit
```

#Configure Device3.

```
Device3#configure terminal
Device3(config)#router ospf 100
Device3(config-ospf)#network 20.1.1.0 0.0.0.255 area 0
Device3(config-ospf)#network 40.1.1.0 0.0.0.255 area 0
Device3(config-ospf)#exit
```

#View the route table of Device1, and you can see that there are two next hops to the network 2.2.2.0/24.

```
Device1#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

C   1.1.1.0/24 is directly connected, 22:14:53, gigabitethernet2
L   1.1.1.1/32 is directly connected, 22:14:53, gigabitethernet2
O   2.2.2.0/24 [110/3] via 10.1.1.2, 00:00:09, gigabitethernet0
               [110/3] via 20.1.1.2, 00:00:09, gigabitethernet1
C   10.1.1.0/24 is directly connected, 21:41:21, gigabitethernet0
L   10.1.1.1/32 is directly connected, 21:41:21, gigabitethernet0
C   20.1.1.0/24 is directly connected, 15:19:15, gigabitethernet1
L   20.1.1.1/32 is directly connected, 15:19:15, gigabitethernet1
O   30.1.1.0/24 [110/2] via 10.1.1.2, 18:55:36, gigabitethernet0
O   40.1.1.0/24 [110/2] via 20.1.1.2, 00:22:08, gigabitethernet1
C   127.0.0.0/8 is directly connected, 87:42:47, lo0
L   127.0.0.1/32 is directly connected, 87:42:47, lo0
```

#Configure Device1, and modify the cost value of the interface gigabitethernet0 to 100 so that the route to the network 2.2.2.0/24 is preferably the gigabitethernet1 interface.

```
Device1(config)#interface gigabitethernet0
Device1(config-if-gigabitethernet0)#ip ospf cost 100
Device1(config-if-gigabitethernet0)#exit
```

#View the route table of Device1.

```
Device1#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

C   1.1.1.0/24 is directly connected, 23:27:34, gigabitethernet2
L   1.1.1.1/32 is directly connected, 23:27:34, gigabitethernet2
O   2.2.2.0/24 [110/3] via 20.1.1.2, 01:12:50, gigabitethernet1
```

```
C   10.1.1.0/24 is directly connected, 22:54:03, gigabitethernet0
L   10.1.1.1/32 is directly connected, 22:54:03, gigabitethernet0
C   20.1.1.0/24 is directly connected, 16:31:57, gigabitethernet1
L   20.1.1.1/32 is directly connected, 16:31:57, gigabitethernet1
O   30.1.1.0/24 [110/3] via 20.1.1.2, 00:31:42, gigabitethernet0
O   40.1.1.0/24 [110/2] via 20.1.1.2, 01:34:50, gigabitethernet1
C   127.0.0.0/8 is directly connected, 88:55:28, lo0
L   127.0.0.1/32 is directly connected, 88:55:28, lo0
```

#View the path to the server 2.2.2.2 through the **traceroute** command on the PC.

```
C:\Documents and Settings\Administrator>tracert 2.2.2.2

Tracing route to 2.2.2.2 over a maximum of 30 hops

  1     1 ms     1 ms     1 ms  1.1.1.2
  2    <1 ms    <1 ms    <1 ms  20.1.1.2
  ......
  n    <1 ms    <1 ms    <1 ms  2.2.2.2
Trace complete.
```

It can be seen that PC accesses server 2.2.2.2 through Device1 and Device3.

Step 3: Configure track to associate BSM, bandwidth utilization and BFD at the same time.

#On Device1, configure BSM.

```
Device1(config)#ntp enable
Device1(config)#ntp master 2
Device1(config)#bsm-id 10.0.0.1
Device1(config)#bsm 1
Device1(config-bsm)#mgt-node
Device1(config-bsm-mgt-node)#stat-node 10.0.0.1
Device1(config-bsm-mgt-node)#stat-node 10.0.0.2
Device1(config-bsm-mgt-node)#measure-level high
Device1(config-bsm-mgt-node)#exit
Device1(config-bsm)#stat-node
Device1(config-bsm-stat-node)#mgt-node 10.0.0.1
Device1(config-bsm-stat-node)#measure-target ip source-address 10.0.0.1
destination-address 10.0.0.2 protocol 0 source-port 0 destination-port 0 dscp 0
Device1(config-bsm-stat-node)#measure-interface gigabitethernet0 send role in
Device1(config-bsm-stat-node)#exit
Device1(config-bsm)#exit
```

#On Device2, configure BSM.

```
Device2(config)#ntp enable
Device2(config)#ntp server 10.0.0.1
Device2(config)#bsm-id 10.0.0.2
Device2(config)#bsm 1
Device2(config-bsm)#stat-node
Device2(config-bsm-stat-node)#mgt-node 10.0.0.1
Device2(config-bsm-stat-node)#measure-target ip source-address 10.0.0.1 destination-address 10.0.0.2 protocol 0 source-port 0 destination-port 0 dscp 0
Device2(config-bsm-stat-node)#measure-interface gigabitethernet0 receive role out
Device2(config-bsm-stat-node)#exit
Device2(config-bsm)#exit
```

#Configure interface bandwidth utilization on Device1.

```
Device1(config)#interface gigabitethernet 0
Device1(config-if-gigabitethernet0)#trap-threshold output-rate 80 resume-rate 50
Device1(config-if-gigabitethernet0)#exit
```

#On Device1, configure track to associate BSM, bandwidth utilization and BFD at the same time.

```
Device1(config)#track 1
Device1(config-track)#bsm 1 deteriorate-degree normal
Device1(config-track)#interface gigabitethernet 0 trap-threshold output
Device1(config-track)#bfd interface gigabitethernet 0 remote-ip 10.0.0.2 local-ip 10.0.0.1
Device1(config-track)#exit
```

#On Device2, configure Track to associate BFD.

```
Device2(config)#track 1
Device2(config-track)#bfd interface gigabitethernet 0 remote-ip 10.0.0.1 local-ip 10.0.0.2
Device2(config-track)#exit
```

Step 3: On Device1, configure PBR to link with TRACK.

#Configure ACL1001, permitting PC to access the network 2.2.2.0/24.

```
Device1(config)#ip access-list extended 1001
Device1(config-ext-nacl)#permit ip any 2.2.2.0 0.0.0.255
Device1(config-ext-nacl)#exit
```

#Configure PBR aaa, associate ACL 1001, specify the next hop as 10.1.1.2, and link track.

```
Device1(config)#route-policy aaa permit 10
Device1(config-pbr)#match ip address acl 1001
```

```
Device1(config-pbr)#set ip next-hop 10.1.1.2 track 1
Device1(config-pbr)#exit
```

#View the information about PBR aaa of Device1.

```
Device1#show route-policy aaa
route-policy aaa
  sequence 10 permit:
    match ip address acl 1001
    set ip next-hop 10.1.1.2 track 1
```

Step 4: Apply PBR.

#On interface gigabitethernet2 of Device1, apply PBR aaa.

```
Device1(config)#interface gigabitethernet2
Device1(config-if-gigabitethernet2)#ip policy aaa
Device1(config-if-gigabitethernet2)#exit
```

Step 5: Check the result.

#On Device1, view the TRACK status.

```
Device1#show track object 1
track 1
 status = up
 entnum = 3
 logic operator AND
 logic reverse FALSE
             Object Type    Status  Refcnt                    instruction
    -------------------------  ---------  ------  ---------------------------------
                  bfd        up       1   bfd interface gigabitethernet0 remote-ip 10.0.0.2
local-ip 10.0.0.1
                  bsm        up       1   bsm 1 deteriorate-degree normal
    interface trap-threshold      up      1   interface gigabitethernet 0 trap-threshold
output
-------------------------------------------------------------------------------
                module  priority  caller
    -------------------------  ---------  ------
                  PBR       20  0x1f5de44
    ---------------------------------------------------------------------------
```

You can see that the status of the TRACK group is up.

#View the path to the server 2.2.2.2 through the traceroute command on the PC.

```
C:\Documents and Settings\Administrator>tracert 2.2.2.2

Tracing route to 2.2.2.2 over a maximum of 30 hops

  1     1 ms     1 ms     1 ms  1.1.1.2
  2    <1 ms    <1 ms    <1 ms  10.1.1.2
  ……
  n    <1 ms    <1 ms    <1 ms  2.2.2.2
Trace complete.
```

You can see that after the PBR is applied to the interface gigabitethernet2, the PC accesses the server 2.2.2.2 through Device1 and Device2.

#When the link between Device1 and Device2 fails, BFD detects down.

View the track status on Device1.

```
Device1#show track object 1
track 1
 status = down
 entnum = 3
 logic operator AND
 logic reverse FALSE
                  Object Type    Status  Refcnt                              instruction
     -------------------------  --------  ------  --------------------------------
                          bfd      down      1   bfd interface gigabitethernet0 remote-ip 10.0.0.2
local-ip 10.0.0.1
                          bsm        up      1   bsm 1 deteriorate-degree normal
      interface trap-threshold       up      1   interface gigabitethernet 0 trap-threshold
output
     ----------------------------------------------------------------------------
                       module  priority  caller
     -------------------------  --------  ------
                          PBR        20  0x1f5de44
     ----------------------------------------------------------------------------
```

You can see that the track group status is down.

#View the path to the server 2.2.2.2 through the traceroute command on the PC.

```
C:\Documents and Settings\Administrator>tracert 2.2.2.2

Tracing route to 2.2.2.2 over a maximum of 30 hops

  1     1 ms     1 ms     1 ms  1.1.1.2
```

```
    2   <1 ms   <1 ms   <1 ms  20.1.1.2
    ……
    n   <1 ms   <1 ms   <1 ms  2.2.2.2
  Trace complete.
```

It can be seen that PC accesses server 2.2.2.2 through Device1 and Device3.

Note:

- Track group can monitor interface status, direct route, route reachability, RTR group, BFD session, etc.

13.3.4. Configure Cross-VPN Forwarding PBR

Network Requirements

- The whole MPLS network includes two VPNs, VPN1 and VPN2.
- PC1 belongs to VPN1, and PC2 belongs to VPN2.
- Configure the cross-VPN forwarding PBR to make PC1 and PC2 communicate with each other.

Network Topology

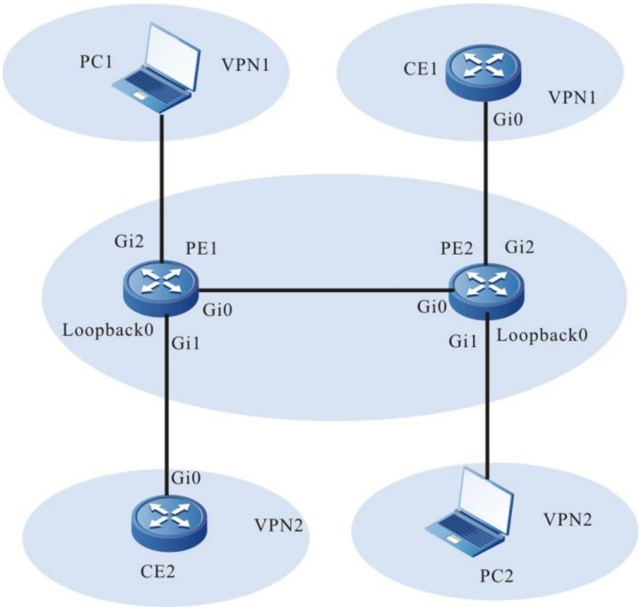

Figure 13–2 Networking for configuring the cross-VPN forwarding PBR

Device	Interface	IP address	Device	Interface	IP address
PC1		9.9.2.2/24	PE2	Gi0	11.1.1.2/24
PE1	Gi0	11.1.1.1/24		Gi1	12.1.3.1/24
	Gi1	9.9.3.1/24		Gi2	12.1.2.1/24

Device	Interface	IP address	Device	Interface	IP address
	Gi2	9.9.2.1/24		Loopback0	22.22.22.22/32
	Loopback0	10.10.10.10/32	CE1	Gi0	12.1.2.2/24
CE2	Gi0	9.9.3.2/24	PC2		12.1.3.2/24

Configuration Steps

Step 1: Configure the interface IP address. CE1, PE1, PE2, and CE2 form the MPLS L3VPN network. PC1 and CE1, PC2 and CE2 can communicate with each other. (omitted, refer to the configuration manual-the MPLS L3VPN chapter)

#View the VPN route table on PE1.

```
PE1#show ip route vrf 1
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

C   9.9.2.0/24 is directly connected, 00:00:21, gigabitethernet2
L   9.9.2.1/32 is directly connected, 00:00:21, gigabitethernet2
B   12.1.2.0/24 [200/0] via 22.22.22.22, 00:19:09, gigabitethernet0

PE1#show ip route vrf 2
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

C   9.9.3.0/24 is directly connected, 00:00:22, gigabitethernet1
L   9.9.3.1/32 is directly connected, 00:00:22, gigabitethernet1
B   12.1.3.0/24 [200/0] via 22.22.22.22, 00:19:10, gigabitethernet0
```

#View the VPN route table on PE2.

```
PE2#show ip route vrf 1
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

B   9.9.2.0/24 [200/0] via 10.10.10.10, 00:02:34, gigabitethernet0
```

```
C   12.1.2.0/24 is directly connected, 00:02:59, gigabitethernet2
L   12.1.2.1/32 is directly connected, 00:02:59, gigabitethernet2

PE2#show ip route vrf 2
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

B   9.9.3.0/24 [200/1] via 10.10.10.10, 00:02:31, gigabitethernet0
C   12.1.3.0/24 is directly connected, 00:03:01, gigabitethernet1
L   12.1.3.1/32 is directly connected, 00:03:01, gigabitethernet1
```

Step 2: Configure the PBR.

#On PE1, configure the ACL 1001, permitting PC1 to access the network 12.1.3.0/24.

```
PE1#configure terminal
PE1(config)#ip access-list extended 1001
PE1(config-ext-nacl)#permit ip any 12.1.3.0 0.0.0.255
PE1(config-ext-nacl)#exit
```

#On PE1, configure the PBR pbr1, and specify the packet of ACL1001 to search for the route forwarding information in VPN1 and VPN2.

```
PE1(config)#route-policy pbr1 permit 10
PE1(config-pbr)#match ip address acl 1001
PE1(config-pbr)#set ip vrf 1 2
PE1(config-pbr)#exit
```

#On PE1, view the information of the PBR pbr1.

```
PE1#show route-policy pbr1
route-policy pbr1
  sequence 10 permit:
    match ip address acl 1001
    set ip vrf 1 2
```

#On PE2, configure ACL 1001, permitting PC2 to access the network 9.9.2.0/24.

```
PE2#configure terminal
PE2(config)#ip access-list extended 1001
PE2(config-ext-nacl)#permit ip any 9.9.2.0 0.0.0.255
PE2(config-ext-nacl)#exit
```

#On PE2, configure the PBR pbr2, and specify the packet of ACL1001 to search for the route forwarding information in VPN1 and VPN2.

```
PE2(config)#route-policy pbr2 permit 10
PE2(config-pbr)#match ip address acl 1001
PE2(config-pbr)#set ip vrf 2 1
PE2(config-pbr)#exit
```

#On PE2, view the information of the PBR pbr2.

```
PE2#show route-policy pbr2
route-policy pbr2
  sequence 10 permit:
    match ip address acl 1001
    set ip vrf 2 1
```

Step 3: Apply the PBR.

#On the interface gigabitethernet2 of PE1, apply the PBR pbr1.

```
PE1(config)#interface gigabitethernet2
PE1(config-if-gigabitethernet2)#ip policy pbr1
PE1(config-if-gigabitethernet2)#exit
```

#On the interface gigabitethernet1 of PE2, apply the PBR pbr2.

```
PE2(config)#interface gigabitethernet1
PE2(config-if-gigabitethernet1)#ip policy pbr2
PE2(config-if-gigabitethernet1)#exit
```

Note:

- After applying the cross-VPN forwarding PBR to PE, the packet matching the ACL searches for the route forwarding information in the specified VPN according to the configuration order. If searching successfully, forward the packet immediately. If failing to search, search in the next VPN. If not finding the route forwarding information in all specified VPNs, drop the packet.

Step 4: Check the result.

#On PC1, ping the address of PC2 12.1.3.2.

```
C:\Documents and Settings\Administrator>ping 12.1.3.2

Pinging 12.1.3.2 with 32 bytes of data:

Reply from 12.1.3.2: bytes=32 time<1ms TTL=125
Reply from 12.1.3.2: bytes=32 time<1ms TTL=125
Reply from 12.1.3.2: bytes=32 time<1ms TTL=125
Reply from 12.1.3.2: bytes=32 time<1ms TTL=125
```

```
Ping statistics for 12.1.3.2:
   Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

You can see that PC1 and PC2 can communicate with each other.

14. IPV6 PBR

14.1. Overview

PBR (policy-based-route) is one technology of forwarding packets according to the configured packet processing policy (including match policy and forwarding policy). When forwarding packets, first match the packet according to the match policy. If matching successfully, forward the packet according to the forwarding policy. The match policy is based on the extended ACL. Forwarding policy includes setting the sending interface of the packet, setting the forwarding next hop of the packet, and so on.

The PBR is divided to forwarding PBR and local PBR:

- Forwarding PBR: It takes effect only for the packet received by the interface that applies the PBR, but does not take effect for the local packet.
- Local PBR: It takes effect only for the local packet, but does not take effect for the forwarded packet.

The priority of the PBR is higher than the common route, that is, the packet first executes the PBR processing. If the packet does not match the PBR match policy, forward according to the common route.

14.2. IPv6 PBR Function Configuration

Table 14-1 IPv6 PBR function configuration list

Configuration Task	
Configure the IPv6 PBR	Create the IPv6 PBR policy and rule
	Configure the match policy
	Configure the forwarding policy
Apply the IPv6 PBR	Apply the forwarding IPv6 PBR
	Apply the local IPv6 PBR

14.2.1. Configure IPv6 PBR

Configuration Condition

None

Create IPv6 PBR Policy and Rule

When creating the IPv6 PBR rule, you should specify the serial number, the smaller the serial number, the higher the rule priority.

Table 14-2 Create the IPv6 PBR policy and rule

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Create the IPv6 PBR policy and rule	**ipv6 route-policy** *policy-name* **permit** *sequence*	Mandatory By default, do not create the IPv6 PBR policy and rule.

Configure Match Policy

The packet executes the PBR operation according to the priority of the IPbv6 PBR rule from high to low. The IPv6 PBR rules include match policy and forwarding policy. If the packet matches successfully, forward the packet according to the forwarding policy, but do not execute the next PBR rule. If the IPv6 PBR rule does not configure the forwarding policy, execute the next rule automatically.

Table 14-3 Configure the match policy

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the IPv6 PBR configuration mode	**ipv6 route-policy** *policy-name* **permit** *sequence*	-
Configure the match policy of the IPv6 packet	**match ipv6 address acl** { *access-list-number* | *access-list-name* }	Optional By default, do not configure the match policy of the IPv6 packet.

Note:

- The match policy of each rule can only associate one ACL.
- When configuring the match policy and if the specified ACL does not exist, the command can be configured successfully, but when all packets fail to match, the match policy does not take effect. The match policy takes effect only after the specified ACL is created.

Configure Forwarding Policy

The device supports four forwarding policies, that is, set the sending next-hop address of the packet, set the sending interface of the packet, set the load balance mode of the packet, drop packet.

- Set the sending next-hop address of the packet: Send the packet to the specified next hop
- Set the sending interface of the packet: Send the packet from the specified interface
- Set the load balance mode of the packet: Adopt the specified load balance mode to forward the packet
- Drop the packet: Directly drop the packet, but do not perform other processing.

Table 14-4 Configure the forwarding policy

Step	**Command**	**Description**
Enter the global configuration mode.	**configure terminal**	-
Enter the IPv6 PBR configuration mode	**ipv6 route-policy** *policy-name* **permit** *sequence*	-
Set the sending next-hop address of the packet	**set ipv6 next-hop** { *ipv6-address-list* } [**track** { *track-id* }]	Optional By default, do not set the sending next-hop address of the packet.
Set the sending interface of the packet	**set interface** { *interface-name-list* } [**track** { *track-id* }]	Optional By default, do not set the sending interface of the packet.
Set the load balance mode of the packet	**set load-sharing** { per-packet | per-source }	Optional By default, do not adopt any load balance mode. Select the first valid forwarding path to forward the packet.
Drop the packet	**set drop**	Optional By default, do not set dropping the packet.

Note:

- The order of executing the operations of setting the sending next-hop address of the packet, setting the sending interface of the packet, and dropping the packet: set the sending next-hop address of the packet > set the sending interface of the packet > drop the packet

- When setting the sending next-hop address of the packet, you can specify one or multiple next hops and the direct-connected next hop is the valid next hop. When specifying multiple next hops, search for the valid next hop according to the configuration order. After finding the valid next hop, send the packet to the next hop.
- When setting the sending interface of the packet, you can specify one or multiple interfaces. The UP interface is the valid sending interface. When specifying multiple sending interfaces, search for the valid sending interface according to the configuration order. After finding the valid sending interface, send the packet from the interface. If the sending interface is Ethernet, only the packet whose destination address is the direct-connected next-hop address of the sending interface can be forwarded successfully. Otherwise, drop the packet.
- Set the load balance mode of the packet: The set sending next hop of the packet and the set sending interface of the packet calculate the load forwarding path respectively, but do not take the sending next hop of the packet and the sending interface of the packet as the load path with the same priority to calculate. Send the packet according to the following order: Set the sending next-hop address of the packet > Set the sending interface of the packet > drop the packet.
- When setting the next hop address or sending interface of the packet to link with the track, the set next hop address or sending interface may become valid only when the track status becomes up.
- The process of setting bound and unbound tracks does not support updating. For example, if **set ipv6 next-hop 1::1** is set first, you need to cancel the setting before successfully setting **set ipv6 next-hop 1::1 track 1**.

14.2.2. Apply IPv6 PBR

Configuration Condition

None

Apply Forwarding PBR

Apply the forwarding IPv6 PBR on the specified interface. The packet received from the interface is forwarded according to the specified IPv6 PBR policy.

Table 14-5 Apply the forwarding IPv6 PBR

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the interface configuration mode.	**interface** *interface-name*	-
Apply the forwarding IPv6 PBR	**ipv6 policy** *policy-name* [**any**]	Mandatory By default, the interface does not apply the forwarding IPv6 PBR.

Note:

- If the applied IPv6 PBR policy of the interface does not exist, the command can be configured successfully, but the packet is forwarded according to the common route. After the applied IPv6 PBR policy of the interface is created, the packet is forwarded according to the IPv6 PBR policy.
- If not configuring any, the interface receives the local packet, but does not execute the PBR processing. If configuring any, the interface receives the local packet and executes the PBR processing.

Apply Local PBR

The local PBR only processes the packet generated by the device.

Table 14-6 Apply the local IP6 PBR

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Apply the local IPv6 PBR	**ipv6 local policy** *policy-name*	Mandatory By default, do not apply the local IPv6 PBR.

Note:

- If the applied IPv6 PBR policy of the interface does not exist, the command can be configured successfully, but the packet generated by the device is forwarded according to the common route. After the applied IPv6 PBR policy is created, the packet is forwarded according to the IPv6 PBR policy.

14.2.3. IPv6 PBR Monitoring and Maintaining

Table 14-7 IPv6 PBR monitoring and maintaining

Command	Description
show ipv6 local route-policy	Display the configuration information of the local IPv6 PBR
show ipv6 policy [*interface-name*]	Display the configuration information of the applied forwarding IPv6 PBR
show ipv6 route-policy [*policy-name*]	Display the configuration information of the IPv6 PBR policy and rule

14.3. IPv6 PBR Typical Configuration Example

14.3.1. Configure Local IPv6 PBR

Network Requirements

- Run the OSPFv3 protocol on all devices and configure the local PBR on Device1.
- There is the server with the IPv6 address 2::2/128 in the IPv6 network.
- Configure the local IPv6 PBR so that Device1 can access the server 2::2 via Device2.

Network Topology

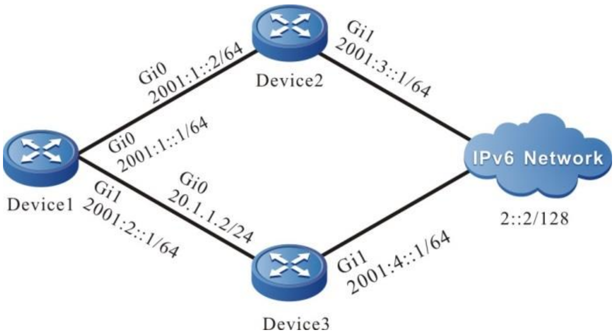

Figure 14-1 Networking for configuring the local IPv6 PBR

Configuration Steps

Step 1: Configure the IPv6 address of the interface (omitted).

Step 2: Enable the unicast route protocol OSPFv3 so that all devices in the network can communicate with each other.

#Configure Device1, configuring the OSPFv3 process and covering the interface to area 0.

```
Device1#configure terminal
Device1(config)#ipv6 router ospf 100
Device1(config-ospf6)#router-id 1.1.1.1
Device1(config-ospf6)#exit
Device1(config)#interface gigabitethernet0
Device1(config-if-gigabitethernet0)#ipv6 router ospf 100 area 0
Device1(config-if-gigabitethernet0)#exit
Device1(config)#interface gigabitethernet1
Device1(config-if-gigabitethernet1)#ipv6 router ospf 100 area 0
Device1(config-if-gigabitethernet1)#exit
```

#Configure Device2, configuring the OSPFv3 process and covering the interface to area 0.

```
Device2#configure terminal
Device2(config)#ipv6 router ospf 100
Device2(config-ospf6)#router-id 2.2.2.2
Device2(config-ospf6)#exit
Device2(config)#interface gigabitethernet0
Device2(config-if-gigabitethernet0)#ipv6 router ospf 100 area 0
```

```
Device2(config-if-gigabitethernet0)#exit
Device2(config)#interface gigabitethernet1
Device2(config-if-gigabitethernet1)#ipv6 router ospf 100 area 0
Device2(config-if-gigabitethernet1)#exit
```

#Configure Device3, configuring the OSPFv3 process and covering the interface to area 0.

```
Device3#configure terminal
Device3(config)#ipv6 router ospf 100
Device3(config-ospf6)#router-id 3.3.3.3
Device3(config-ospf6)#exit
Device3(config)#interface gigabitethernet0
Device3(config-if-gigabitethernet0)#ipv6 router ospf 100 area 0
Device3(config-if-gigabitethernet0)#exit
Device3(config)#interface gigabitethernet1
Device3(config-if-gigabitethernet1)#ipv6 router ospf 100 area 0
Device3(config-if-gigabitethernet1)#exit
```

#View the route table of Device1 and you can see that there are two next hops to the network 2::2/128.

```
Device1#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management


L   ::1/128 [0/0]
    via ::, 1d:21:18:04, lo0
O   2::2/128 [110/3]
    via fe80::201:7aff:fe5e:6d45, 00:01:51, gigabitethernet0
         [110/3]
    via fe80::201:7aff:fe00:105, 00:01:41, gigabitethernet1
C   2001:1::/64 [0/0]
    via ::, 00:15:48, gigabitethernet0
L   2001:1::1/128 [0/0]
    via ::, 00:15:44, lo0
C   2001:2::/64 [0/0]
    via ::, 00:02:22, gigabitethernet1
L   2001:2::1/128 [0/0]
    via ::, 00:02:20, lo0
O   2001:3::/64 [110/2]
```

```
   via fe80::201:7aff:fe5e:6d45, 00:01:51, gigabitethernet0
O  2001:4::/64 [110/2]
   via fe80::201:7aff:fe00:105, 00:01:41, gigabitethernet1
```

#Configure Device1, and modify the cost value of the interface gigabitethernet0 to 100 to make the route to the network 2::2/128 first select the interface gigabitethernet1.

```
Device1(config)#interface gigabitethernet0
Device1(config-if-gigabitethernet0)#ipv6 ospf cost 100
Device1(config-if-gigabitethernet0)#exit
```

#View the route table of Device1.

```
Device1#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
     U - Per-user Static route
     O - OSPF, OE-OSPF External, M - Management

L  ::1/128 [0/0]
   via ::, 1d:21:18:26, lo0
O  2::2/128 [110/3]
   via fe80::201:7aff:fe00:105, 00:02:03, gigabitethernet1
C  2001:1::/64 [0/0]
   via ::, 00:16:10, gigabitethernet0
L  2001:1::1/128 [0/0]
   via ::, 00:16:07, lo0
C  2001:2::/64 [0/0]
   via ::, 00:02:44, gigabitethernet1
L  2001:2::1/128 [0/0]
   via ::, 00:02:43, lo0
O  2001:3::/64 [110/3]
   via fe80::201:7aff:fe00:105, 00:00:00, gigabitethernet1
O  2001:4::/64 [110/2]
   via fe80::201:7aff:fe00:105, 00:02:03, gigabitethernet1
```

#On Device1, use the Traceroute command to view the path to the server 2::2.

```
Device1#traceroute 2::2

Type escape sequence to abort.
Tracing the route to 2::2 , min ttl = 1, max ttl = 30 .

  1  2001:2::2 0 ms    0 ms    0 ms
  ......
   n  2::2 0 ms    0 ms    0 ms
```

You can see that Device1 access the server 2::2 via Device3.

Step 3: Configure the PBR on Device1.

#Configure IPv6 ACL 7001, permitting the device to access the network 2::2/128.

```
Device1(config)#ipv6 access-list extended 7001
Device1(config-v6-list)#permit ipv6 any host 2::2
Device1(config-v6-list)#exit
```

#Configure IPv6 PBR aaa, associate with the ACL 7001, and specify the next hop as 2001:1::2.

```
Device1(config)#ipv6 route-policy aaa permit 10
Device1(config-pbr6)#match ipv6 address acl 7001
Device1(config-pbr6)#set ipv6 next-hop 2001:1::2
Device1(config-pbr6)#exit
```

#View the information of Device1 IPv6 PBR aaa.

```
Device1#show ipv6 route-policy aaa
ipv6 route-policy aaa
  sequence 10 permit:
    match ipv6 address acl 7001
    set ipv6 next-hop 2001:1::2
```

Step 4: Apply the IPv6 PBR.

#Apply the local IPv6 PBR aaa on Device1.

```
Device1(config)#ipv6 local policy aaa
```

Step 5: Check the result.

#On Device1, use the Traceroute command to view the path to the server 2::2.

```
Device1#traceroute 2::2

Type escape sequence to abort.
Tracing the route to 2::2 , min ttl = 1, max ttl = 30 .

  1  2001:1::2  0 ms    0 ms    0 ms
  ......
   n  2::2  0 ms    0 ms    0 ms
```

You can see that after applying the local IPv6 PBR, Device1 accesses the server 2::2 via Device2.

14.3.2. Configure Forwarding IPv6 PBR

Network Requirements

- Run the OSPFv3 protocol on all devices, making all devices in the network communicate with each other.
- There is the server with the IPv6 address 2::2/128 in the IPv6 network.
- Configure the forwarding IPv6 PBR on Device1 so that PC can access the server 2::2 via Device1, Device2.

Network Topology

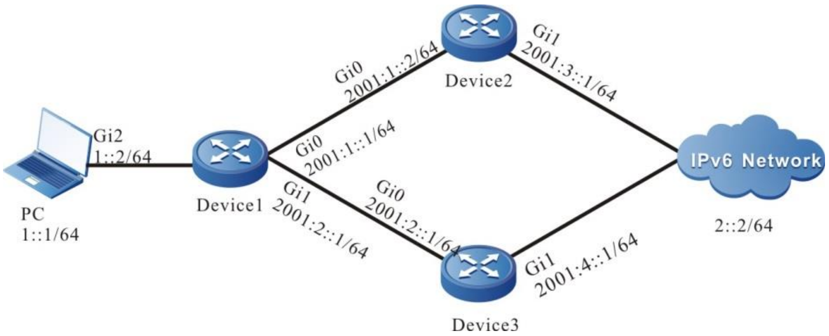

Figure 14–1 Networking for configuring the forwarding IPv6 PBR

Configuration Steps

Step 1: Configure the IPv6 address of the interface (omitted).

Step 2: Enable the unicast route protocol OSPFv3 so that all devices in the network can communicate with each other.

#Configure Device1, configuring the OSPFv3 process and covering the interface to area 0.

```
Device1#configure terminal
Device1(config)#ipv6 router ospf 100
Device1(config-ospf6)#router-id 1.1.1.1
Device1(config-ospf6)#exit
Device1(config)#interface gigabitethernet0
Device1(config-if-gigabitethernet0)#ipv6 router ospf 100 area 0
Device1(config-if-gigabitethernet0)#exit
Device1(config)#interface gigabitethernet1
Device1(config-if-gigabitethernet1)#ipv6 router ospf 100 area 0
Device1(config-if-gigabitethernet1)#exit
Device1(config)#interface gigabitethernet2
Device1(config-if-gigabitethernet2)#ipv6 router ospf 100 area 0
Device1(config-if-gigabitetherne2)#exit
```

#Configure Device2, configuring the OSPFv3 process and covering the interface to area 0.

```
Device2#configure terminal
Device2(config)#ipv6 router ospf 100
Device2(config-ospf6)#router-id 2.2.2.2
```

```
Device2(config-ospf6)#exit
Device2(config)#interface gigabitethernet0
Device2(config-if-gigabitethernet0)#ipv6 router ospf 100 area 0
Device2(config-if-gigabitethernet0)#exit
Device2(config)#interface gigabitethernet1
Device2(config-if-gigabitethernet1)#ipv6 router ospf 100 area 0
Device2(config-if-gigabitethernet1)#exit
```

#Configure Device3, configuring the OSPFv3 process and covering the interface to area 0.

```
Device3#configure terminal
Device3(config)#ipv6 router ospf 100
Device3(config-ospf6)#router-id 3.3.3.3
Device3(config-ospf6)#exit
Device3(config)#interface gigabitethernet0
Device3(config-if-gigabitethernet0)#ipv6 router ospf 100 area 0
Device3(config-if-gigabitethernet0)#exit
Device3(config)#interface gigabitethernet1
Device3(config-if-gigabitethernet1)#ipv6 router ospf 100 area 0
Device3(config-if-gigabitethernet1)#exit
```

#View the route table of Device1 and you can see that there are two next hops to the network 2::2/128.

```
Device1#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
     U - Per-user Static route
     O - OSPF, OE-OSPF External, M - Management

L   ::1/128 [0/0]
    via ::, 00:07:41, lo0
C   1::2/128 [0/0]
    via ::, 00:00:40, gigabitethernet2
O   2::2/128 [110/3]
    via fe80::201:7aff:fe00:105, 00:04:36, gigabitethernet0
          [110/3]
    via fe80::201:7aff:fe5e:6d45, 00:04:36, gigabitethernet1
C   2001:1::/64 [0/0]
    via ::, 00:06:45, gigabitethernet0
L   2001:1::1/128 [0/0]
    via ::, 00:06:42, lo0
```

```
C   2001:2::/64 [0/0]
    via ::, 00:06:46, gigabitethernet1
L   2001:2::1/128 [0/0]
    via ::, 00:06:43, lo0
O   2001:3::/64 [110/2]
    via fe80::201:7aff:fe5e:6d45, 00:04:36, gigabitethernet0
O   2001:4::/64 [110/2]
    via fe80::201:7aff:fe00:105, 00:04:36, gigabitethernet1
```

#Configure Device1, and modify the cost value of the interface gigabitethernet0 to 100 to make the route to the network 2::2/128 first select the interface gigabitethernet1.

```
Device1(config)#interface gigabitethernet0
Device1(config-if-gigabitethernet0)#ipv6 ospf cost 100
Device1(config-if-gigabitethernet0)#exit
```

#View the route table of Device1.

```
Device1#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
      U - Per-user Static route
      O - OSPF, OE-OSPF External, M - Management

L   ::1/128 [0/0]
    via ::, 00:10:39, lo0
C   1::2/128 [0/0]
    via ::, 00:03:37, gigabitethernet2
O   2::2/128 [110/3]
    via fe80::201:7aff:fe00:105, 00:07:33, gigabitethernet1
C   2001:1::/64 [0/0]
    via ::, 00:09:42, gigabitethernet0
L   2001:1::1/128 [0/0]
    via ::, 00:09:39, lo0
C   2001:2::/64 [0/0]
    via ::, 00:09:43, gigabitethernet0
L   2001:2::1/128 [0/0]
    via ::, 00:09:40, lo0
O   2001:3::/64 [110/3]
    via fe80::201:7aff:fe00:105, 00:00:07, gigabitethernet1
O   2001:4::/64 [110/2]
    via fe80::201:7aff:fe00:105, 00:07:33, gigabitethernet1
```

#On PC, use the Traceroute command to view the path to the server 2::2.

```
C:\Documents and Settings\Administrator>tracert 2::2

Tracing route to 2::2 over a maximum of 30 hops


  1    1 ms     1 ms     1 ms  1::2
  2   <1 ms    <1 ms    <1 ms  2001:2:2
  ……
  n   <1 ms    <1 ms    <1 ms  2::2
Trace complete.
```

You can see that PC accesses the server 2::2 via Device1, Device3.

Step 3: Configure the IPv6 PBR on Device1.

#Configure IPv6 ACL 7001, permitting the device to access the network 2::2/128.

```
Device1(config)#ipv6 access-list extended 7001
Device1(config-v6-list)#permit ipv6 any host 2::2
Device1(config-v6-list)#exit
```

#Configure IPv6 PBR aaa, associate with the ACL 7001, and specify the next hop as 2001:1::2.

```
Device1(config)#ipv6 route-policy aaa permit 10
Device1(config-pbr6)#match ipv6 address acl 7001
Device1(config-pbr6)#set ipv6 next-hop 2001:1::2
Device1(config-pbr6)#exit
```

#View the information of Device1 IPv6 PBR aaa.

```
Device1#show ipv6 route-policy aaa
ipv6 route-policy aaa
  sequence 10 permit:
    match ipv6 address acl 7001
    set ipv6 next-hop 2001:1::2
```

Step 4: Apply the IPv6 PBR.

#On the interface gigabitethernet2 of Device1, apply the IPv6 PBR aaa.

```
Device1(config)#interface gigabitethernet2
Device1(config-if-gigabitethernet2)#ipv6 policy aaa
Device1(config-if-gigabitethernet2)#exit
```

Step 5: Check the result.

#On PC, use the Traceroute command to view the path to the server 2::2.

```
C:\Documents and Settings\Administrator>tracert 2::2
```

```
Tracing route to 2::2 over a maximum of 30 hops

  1     1 ms     1 ms     1 ms  1::2
  2    <1 ms    <1 ms    <1 ms  2001:1:2
  ……
  n    <1 ms    <1 ms    <1 ms  2::2
Trace complete.
```

You can see that PC accesses the server 2::2 via Device1, Device2.

15. PBR TOOLS

15.1. Overview

A routing policy can change properties or reachability of a route so as to change the routing information or change the paths that the data flow passes. A routing policy is mainly applied in the following aspects:

- Sets route properties: Sets the required route properties for the routes that match the routing policy.
- Controls route advertisement: When a routing protocol advertises route, it advertises only the routes that meet the requirements.
- Controls route receiving: When a routing protocol, it receives only the routes that meet the requirements so as to control the number of routes and improves the network security.
- Controls route redistribution: When a routing protocol redistributes external routes, it introduces only the routes that meet the requirements. A routing policy tool can also be used to set some properties for the external routes that are introduced.

Key-chain is a password management tool. It provides authentication passwords for the routing protocol to authentication protocol packets.

15.2. Configure PBR Tools

Table 15-1 Routing policy tool list

Configuration Tasks	
Configure a prefix list.	Configure a prefix list.
Configure an AS-PATH list.	Configure an AS-PATH list.
Configure a Community-list.	Configure a Community-list.
Configure an Extcommunity-list.	Configure an Extcommunity-list.
Configure a route map.	Create a route map.
	Configure the match clauses of a route map.
	Configure the set clauses of a route map.
Configure a key chain.	Configure a key chain.

15.2.1. Configure Prefix List

Configuration Condition

None

Configure a Prefix List

The prefix list filters routes based on prefixes. The ACL is first designed to filter data packets and then used to filter routes while the prefix list is designed to filter routes. Through some route filtering functions of the ACL and prefix list are the same, the prefix list is more flexible than the ACL.

A prefix list is identified by a prefix list name. Each prefix list contains multiple entries, and each entry can specify a matching range independently. Each entry has a serial number, indicating the sequence in which the prefix list implements matching checks.

The entries of a prefix list are in the OR relation. When a route tries to match a prefix list, it checks the entries in the sequence of small to large. Once the route matches an entry, it passes the filtration of the prefix list, and the next entry will no longer be checked.

1. Configure IPv4 prefix list

Table 15-2 Configure an IPv4 prefix list

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Configure an IPv4 prefix list.	**ip prefix-list** *prefix-list-name* [**seq** *seq-value*] { **deny |** **permit** } *network / length* [**ge** *ge-value*] [**le** *le-value*]	Mandatory. By default, no IPv4 prefix list is configured.

Note:

- The value range is 0<=length<ge-value<=le-value<=32, where "ge" means equal to or larger than, and "le" means equal to or smaller than. If **ip prefix-list** test **permit** 192.168.0.0/16 **ge** 18 **le** 24 is configured, it indicates that routes with the address 192.168.0.0 and mask length of 18 to 24 (including 18 and 24) are allowed to pass.
- If network/length is set to 0.0.0.0/0, it means to match the default route. If 0.0.0.0/0 **le** 32 is configured, it means to match all routes.
- If an implicit expression is contained at the end of an IPv4 prefix list, it means to deny all entries: **deny** 0.0.0.0/0 **le** 32. If you want to deny some routes by configuring a deny statement, it is recommended that you add a **permit** 0.0.0.0/0 **le** 32 statement to allow other IPv4 routes to pass.

1. Configure IPv6 prefix list

Table 15-3 Configure an IPv6 prefix list

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Configure an IPv6 prefix list.	**ipv6 prefix-list** *prefix-list-name* [**seq** *seq-value*] { **deny |** **permit** } *network / length* [**ge** *ge-value*] [**le** *le-value*]	Mandatory. By default, no IPv6 prefix list is configured.

Note:

- The value range is 0<=length<ge-value<=le-value<=32, where "ge" means equal to or larger than, and "le" means equal to or smaller than. If ipv6 prefix-list test permit 100::/16 ge 18 le 24 is configured, it indicates that routes with the address 100::/16 and mask length of 18 to 24 (including 18 and 24) are allowed to pass.
- If network/length is set to 0::/0, it means to match the default route. If 0::/0 le 128 is configured, it means to match all routes.
- If an implicit expression is contained at the end of an IPv6 prefix list, it means to deny all entries: deny 0::/0 le 128. If you want to deny some routes by configuring a deny statement, it is recommended that you add a permit 0::/0 le 128 statement to allow other IPv6 routes to pass.

15.2.2. Configure AS-PATH List

Configuration Condition

None

Configure an AS-PATH List

An AS-PATH list is a tool for filtration based on AS numbers. It is used for BGP route filtration. The AS path property of a BGP route records all ASs that the route passes. When BGP advertises a route to a network outside the local AS, it adds the local AS number to the AS path property to record the AS paths that the route passes.

An AS-PATH list contains multiple entries, and the entries are in the OR relation. When a route tries to match an AS-PATH list, it checks the entries following the sequence of configuration. Once the route matches an entry, it passes the filtration of the AS-PATH list.

Table 15-4 Configure an AS-PATH list

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Configure an AS-PATH list.	**ip as-path access-list** *path-list-number* { **permit** | **deny** } *regular-expression*	Mandatory. By default, no AS-PATH list is configured.

An AS-PATH list uses a regular expression to specify a collection of AS properties that meet the requirement. A regular expression consists of some common characters and some metacharacters. Common characters includes upper- and lower-case characters and numbers while metacharacters have special meanings, as shown in the following table.

Table 15-5 Meanings of metacharacters in a regular expression

Symbol	Meaning
.	Matches any single character.
*	Matches a sequence which consists of 0 or more bits in the mode.
+	Matches a sequence which consists of 1 or more bits in the mode.
?	Matches a sequence which consists of 0 or 1 bit in the mode.
^	Matches the start of the inputted character string.
$	Matches the end of the inputted character string.
_	Matches commas, brackets, start and end of the inputted character string, and blank spaces.
[]	Matches single characters in a certain range.
-	Separates the end point of a range.

15.2.3. Configure Community-List

Configuration Condition

None

Configure a Community-List

A Community-list is used to filter community properties of routes. Usually, a route consists of two parts: prefix and routing properties. Routing properties are different for different routing protocols. The IGP protocol usually provides simple properties such as metric, but the BGP protocol provides complex properties such as community property. A Community-list is used for filtration.

Filtration on a Community-list acts on the route on which the community property is configured. That is, if the filtration result is deny, the route instead of the community property is filtered.

Two types of Community-lists are available: standard Community-list and extended Community-list. A standard Community-list filters BGP routes based on the local-AS, internet, no-advertise, no-export properties. An extended Community-list filters BGP routes with community properties based on a regular expression.

A Community-list can be used for a routing protocol with community properties. However, you need to bind the Community-list with a route map, and then apply the route map to the routing protocol.

A Community-list contains multiple entries, and the entries are in the OR relation. When a route tries to match a Community-list, it checks the entries following the sequence of configuration. Once the route matches an entry of the community list, it passes the filtration of the community list. For the use of a regular expression in configuring an extended Community-list, refer to "Configure an AS-PATH List".

Table 15-6 Configure a community-list

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Configure a standard Community-list.	**ip community-list** { *community-list-number* | **standard** *community-list-name* } { **permit** | **deny** } [*community-number* / *aa*:*nn* / **local-AS** / **internet** / **no-advertise** / **no-export**]	Mandatory. By default, no standard Community-list is configured.
Configure an extended Community-list.	**ip community-list** { *community-list-number* | **expanded** *community-list-name* } { **permit** | **deny** } *regular-expression*	Mandatory. By default, no extended Community-list is configured.

15.2.4. Configure Extcommunity-List

Configuration Condition

None

Configure an Extcommunity-List

An extended community list (Extcommunity-list) filters BGP routes based on the extended community properties. The quality and usage method of an extended community list (Extcommunity-list) are the same as a standard community list. The major difference is that extended community properties are mainly used in a Multi Protocol Label Switching (MPLS) Layer 3 Virtual Private Network (L3VPN), so an Extcommunity list is also mainly used in an MPLS L3VPN.

Two types of Extcommunity-lists are available: standard Extcommunity-list and extended Extcommunity-list. The standard Extcommunity-list filters BGP routes based on Router Target and Site or Origin properties. An extended Extcommunity-list filters BGP routes with community properties based on a regular expression.

An Extcommunity-list contains multiple entries, and the entries are in the OR relation. When a route tries to match an Extcommunity-list, it checks the entries following the sequence of configuration. Once the route matches an entry, it passes the filtration of the Extcommunity-list. For the use of a regular expression in configuring an extended Extcommunity-list, refer to "Configure an AS-PATH List".

Table 15-7 Configure an extcommunity-list

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Configure a standard Extcommunity-list.	**ip extcommunity-list** { *extcommunity-list-number* | **standard** *extcommunity-list-name*} { **permit | deny** } [**rt** *extcommunity-number* / **soo** *extcommunity-number*]	Mandatory. By default, no standard Extcommunity-list is configured.
Configure an extended Extcommunity list.	**ip extcommunity-list** { *extcommunity-list-number* **|** **expanded** *extcommunity-list-name* } { **permit | deny** } *regular-expression*	Mandatory. By default, no extended Extcommunity-list is configured.

15.2.5. Configure Route Map

A route map is a tool for matching routes and setting route properties. A route map consists of multiple statements, and each statement consists of some match clauses and set clauses. The match clauses define the matching rules of the statement, and the set clauses define the followup actions after a route match the match clauses. The match clauses are in the OR relation, that is, a route must match all match clauses of the statement.

The route map statements are in the OR relation. When a route tries to match a route map, it checks the entries in the sequence of small to large. Once a route matches a statement, it matches the route map, and the next statement will no longer be checked. If a route fails to match a statement, it fails to match the route map.

Configuration Condition

Before configuring a route map, ensure that:

- The ACL, prefix list, AS-PATH, and Community-list or Extcommunity-list that are required for configuring a route map have been configured.

Create a Route Map

In creating a route map, you can specify the match mode of the statements of the route map. Two match modes are available: **permit** and **deny**.

The **permit** mode sets the matching mode of the statements of the route map to permit, that is, if a route matches all match clauses of the statement, the route is allowed to pass, and then the set clauses of the statement are executed. If a route fails to match the match clauses of the statement, it starts to match the next statement of the route map.

The **deny** mode sets the matching mode of the statements of the route map to deny, that is, when a route matches all match clauses of a statement, the route is denied, and the route will not match the next statement of the route map. In **deny** mode, set clauses will not be executed.

Table 15-8 Create a route map

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Create a route map.	**route-map** *map-name* [{ **permit** | **deny** } [*seq-number*]]	Mandatory. By default, no route map is created.

Note:

- If you run the **route-map** command to create a route map, if you configure only the route map name but do not configure the match mode and statement serial number, a statement whose match mode is permit and serial number is 10 is automatically created.
- If a route map is applied to the routing protocol but the route map has not been configured, all objects will fail to match.

Configure the Match Clauses of a Route Map

The match clauses of a route map statement are in the OR relation, that is, a route must match all match clauses before it is allowed to pass.

Table 15-9 Configure the match clause of a route map

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the route map configuration mode.	**route-map** *map-name* [{ **permit** | **deny** } [*seq-number*]]	-
Specify the AS-PATH list that the route map matches.	**match as-path** *path-list-number*	Optional. By default, no AS-PATH list that the route map matches is specified.
Specify the BGP Community-list that the route map matches.	**match community** *community-list-number* / *community-list-name* [**exact-match**]	Optional. By default, no BGP Community-list that the route map matches is specified.
Specify the BGP Extcommunity-list that the route map matches.	**match extcommunity** ext*community-list-number* / ext*community-list-name*	Optional. By default, no BGP Extcommunity-list that the route map matches is specified.
Specify the interface that the route map matches.	**match interface** *interface-names*	Optional. By default, no interface that the route map matches is specified.
Specify the IPv4 route prefix that the route map matches.	**match ip address** { *access-list-number* | *access-list-name* | **prefix-list** *prefix-list-name* }	Optional. By default, no route prefix that the route map matches is specified.

Step	Command	Description
Specify the IPv6 route prefix that the route map matches.	**match ipv6 address** { *access-list-number* | *access-list-name* | **prefix-list** *prefix-list-name* }	Optional. By default, no route prefix that the route map matches is specified.
Specify the next-hop IPv4 address that the route map matches.	**match ip next-hop** { *access-list-name* | **prefix-list** *prefix-list-name* }	Optional. By default, no next-hop address that the route map matches is specified.
Specify the next-hop IPv6 address that the route map matches.	**match ipv6 next-hop** { *access-list-number* | *access-list-name* | **prefix-list** *prefix-list-name* }	Optional. By default, no next-hop address that the route map matches is specified.
Specify the IPv4 source route address that the route map matches.	**match ip route-source** { *access-list-name* | **prefix-list** *prefix-list-name* }	Optional By default, no source route address that the route map matches is specified.
Specify the IPv6 source route address that the route map matches.	**match ipv6 route-source** { *access-list-number* | *access-list-name* | **prefix-list** *prefix-list-name* }	Optional. By default, no source route address that the route map matches is specified.
Specify the route metric value that the route map matches.	**match metric** *metric-value* [**+-***offset*]	Optional. By default, no route metric value that the route map matches is specified.
Specify the routing type that the route map matches.	**match route-type** { **external** / **interarea** / **internal** / **level-1** / **level-2** / **nssa-external** / **type-1** / **type-2** }	Optional. By default, no routing type that the route map matches is specified.

Step	Command	Description
Specify the tag value that the route map matches.	**match tag** *tag-value*	Optional. By default, no tag value that the route map matches is specified.

Note:

- If a route map is not configured with match clauses, all objects can match the route map successfully.
- When the ACL and prefix list that are associated with the match clauses do not exist, no object can match the route map.

Configure the Set Clauses of a Route Map

When the route map match mode is permit, if a route matches all match clauses, the set operations will be executed. If the match mode is deny, the set operations will not be performed.

Table 15-10 Configure the set clauses of a route map

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Enter the route map configuration mode.	**route-map** *map-name* [{ **permit** | **deny** } [*seq-number*]]	-
Set the AS path property of a BGP route.	**set as-path { prepend | replace }** *as-path-number*	Optional. By default, the AS path property of the BGP route is not configured.
Configure the community property of the BGP route.	**set communtiy** { *community-number* | **additive** | **local-AS** | **internet** | **no-advertise** | **no-export** | **none** }	Optional. By default, the community property of the BGP route is not configured.
Delete the Community-list of the BGP route.	**set comm-list** { *community-list-number* / *community-list-name* } **delete**	Optional. By default, the community property of the BGP route is not deleted.

Step	Command	Description
Set BGP route attenuation parameters.	**set dampening** *half-life start-reusing start-suppress max-duration*	Optional. By default, BGP route attenuation parameters are not set.
Set Extcommunity properties of the MPLS L3VPN route.	**set extcommunity** { **rt** | **soo** } *extcommunity*	Optional. By default, the Extcommunity properties of MPLS L3VPN are not configured.
Set the IPv4 route next hop	**set ip default next-hop** *ip-address*	Optional. By default, do not set the route next hop. When being used for the OSPF route re-distribution, set the route next hop.
Set the IPv6 route next hop	**set ipv6 default next-hop** *ipv6-address*	Optional. By default, do not set the route next hop. When being used for the OSPFv3 route re-distribution, set the route next hop.
Set the IPv4 route next hop	**set ip next-hop** *ip-address*	Optional. By default, do not set the route next hop. When being used by BGP to associate the route map, set the route next hop.

Step	Command	Description
Set the IPv6 route next hop	**set ipv6 next-hop** *ipv6-address*	Optional. By default, do not set the route next hop. When being used by IPv6 BGP to associate the route map, set the route next hop.
Set the IS-IS redistribution route type	**set level** { **level-1** | **level-1-2** | **level-2** }	Optional. By default, do not configure the IS-IS redistribution route type.
Set the local priority of BGP route.	**set local-preference** *value*	Optional. By default, the local priority is not configured for the BGP route.
Set the metric value of the route.	**set metric** { *metric* | *+metric* | *-metric* | *bandwidth delay reliable loading mtu* }	Optional. By default, the metric value of the route is not configured.
Set the metric type of the route.	**set metric-type** { **external** | **internal** | **type-1** | **type-2** }	Optional. By default, the metric type of the route is not configured.
Configure the Origin property of the BGP route.	**set origin** { **egp** *as-number* | **igp** | **incomplete** }	Optional. By default, the Origin property of the BGP route is not configured.
Set the tag option field of external routes.	**set tag** *tag-value*	Optional. By default, the tag option field of external routes is not configured.

Step	Command	Description
Set the weight of the BGP route.	**set weight** *weight-value*	Optional. By default, the weight of the BGP route is not configured.

15.2.6. Configure Key Chain

Key chain is a password management tool. It provides authentication passwords for the routing protocol to authentication protocol packets. A key chain provides different passwords for transmitting and receiving packets, and it provides different passwords for different Key IDs. Meanwhile, a key chain can automatically switch passwords according to the validity duration of keys, that is, it uses different keys in different periods of time. This greatly enhances the password security.

Configuration Condition

None

Configure a Key Chain

You can configure multiple Key IDs for a key chain. When a protocol uses the key chain for authentication, it obtains the Key ID according to the following rules:

- The minimum valid transmit passwords of the Key IDs are obtained as the transmit passwords.
- Among the Key IDs that are larger than the specified key IDs of the protocol, obtain the minimum valid receive passwords of the Key IDs as the receive passwords.
- If a Key ID is contained in the received protocol packets, a search for the valid receive passwords are performed based on the Key ID. Otherwise, the minimum valid receive passwords of the Key IDs in the local key chain is used as the receive password.

Table 15-11 Configure a key chain

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Configure a key chain.	**key chain** *keychain-name*	Mandatory. By default, the key chain is not configured.
Configure a Key ID.	**key** *key-id*	Mandatory. By default, the key ID is not configured.

Step	Command	Description
Configure a password.	**key-string** [**0 | 7**] *password*	Mandatory. By default, no password is configured. A blank space is also regarded as a password character. Pay attention to this while configuring a password.
Configure the valid duration in which a key acts as the receive password.	**accept-lifetime** { *time-start* { *time-end* | **duration** *second* | **infinite** } **|** **daily**{*time-range-srting*}| **day**{**sun-sat**}| **date**{1-31}| **month**{**JAN-DEC**}**}**	Mandatory. By default, the receive password is always valid.
Configure the valid duration in which a key acts as the transmit password.	**send-lifetime** { *time-start* { *time-end* | **duration** *second* | **infinite** } **|** **daily**{*time-range-srting*}| **day**{**sun-sat**}| **date**{1-31}| **month**{**JAN-DEC**}**}**	Mandatory. By default, the transmit password is always valid.
Set authentication algorithm	**algorithm {simple|md5|sm3}**	Mandatory By default, do not set authentication alhorithm.

Configure Key Chain Parameters

You can configure the parameters, such as fault tolerance time, TCP authentication algorithm and periodic mode, for a key chain.

Table 15-12 Configure the key chain parameters

Step	Command	Description
Enter the global configuration mode.	**configure terminal**	-
Configure a key chain.	**key chain** *keychain-name*	Mandatory. By default, the key chain is not configured.

Step	Command	Description
Configure the accept tolerance time of Keychain	**accept-tolerance {min<1-14400>| infinite }**	Mandatory By default, do not set the accept tolerance time of Keychain.
Configure the type value in TCP enhanced authentication options	**tcp-kind** *kind-value*	Optional By default, the TCP type value of the Keychain application is 254.
Set the algorithm ID of the TCP authentication algorithm supported by Keychain	**tcp-algorithm-id {md5|sm3}** *algorithm-id*	Mandatory By default, do not set the algorithm ID of the TCP authentication algorithm supported by Keychain.
Set the periodic time mode of keychain	**mode periodic {daily| weekly | monthly |yearly }**	Mandatory By default, it is the default time mode (absolute time mode).

15.2.7. Routing Policy Monitoring and Maintaining

Table 15-13 Routing policy monitoring and maintaining

Command	Description
clear ip prefix-list [*prefix-list-name network/length*]	Clears the prefix list statistics.
show ip prefix-list [*prefix-list-name* [*network/lenghter* [**first-match | longer**] | **seq** *sep_value*] | **detail** [*prefix-list-name*] | **orf-prefix | summary** [*prefix-list-name*]	Display the information about a prefix list.
show ip as-path-access-list [*list-name*]	Display the information about an AS-PATH list.

Command	Description
show ip community-list [*community-list-number* | *community-list-name*]	Display the information about a Community list.
show ip extcommunity-list [*extcommunity-list-number* | *extcommunity-list-name*]	Display the information about an Extcommunity list.
clear ipv6 prefix-list [*prefix-list-name network/length*]	Clear the statistics information of the IPv6 prefix list
show ipv6 prefix-list [*prefix-list-name* [*network/lenghter* [**first-match | longer**] | **seq** *sep_value*] | **detail** [*prefix-list-name*] | **orf-prefix | summary** [*prefix-list-name*]	Display the IPv6 prefix list information
show route-map [*route-map-name*]	Display the information about a route map.
show key chain [*keychain-name*]	Display the information about a key chain.

15.3. PBT Tool Typical Configuration Example

15.3.1. Configure Route Redistribution with the Routing Policy

Network Requirements

- Run OSPF between Device1 and Device2, and run RIP between Device2 and Device3.
- On Device2, configure OSPF to redistribute RIP routes, and associate a routing policy to modify route properties. It is required that the tag property of route 100.1.1.0/24 is changed to 5, the metric value of route 110.1.1.0/24 is changed to 50, and the property of route 120.1.1.0/24 keeps unchanged.

Network Topology

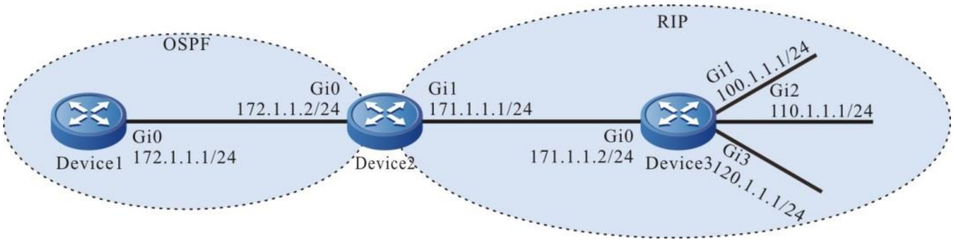

Figure 15–1 Configuring the route redistribution and associating the routing policy

Configuration Steps

Step 1: Configure the IP addresses of the interfaces. (Omitted)

Step 2: Configure OSPF.

#Configure Device1.

```
Device1#configure terminal
Device1(config)#router ospf 100
Device1(config-ospf)#network 172.1.1.0 0.0.0.255 area 0
Device1(config-ospf)#exit
```

#Configure Device2.

```
Device2#configure terminal
Device2(config)#router ospf 100
Device2(config-ospf)#network 172.1.1.0 0.0.0.255 area 0
Device2(config-ospf)#exit
```

Step 3: Configure RIP.

#Configure Device2.

```
Device2(config)#router rip
Device2(config-rip)#version 2
Device2(config-rip)#network 171.1.1.0
Device2(config-rip)#exit
```

#Configure Device3.

```
Device3(config)#configure terminal
Device3(config)#router rip
Device3(config-rip)#version 2
Device3(config-rip)#network 171.1.1.0
Device3(config-rip)#network 100.1.1.0
Device3(config-rip)#network 110.1.1.0
Device3(config-rip)#network 120.1.1.0
Device3(config-rip)#exit
```

Step 4: Configure OSPF to redistribute RIP routes.

#Configure Device2.

```
Device2(config)#router ospf 100
Device2(config-ospf)#redistribute rip
Device2(config-ospf)#exit
```

#Query the route table of Device1.

```
Device1#show ip route
```

```
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

OE  100.1.1.0/24 [150/20] via 172.1.1.2, 02:22:08, gigabitethernet0
OE  110.1.1.0/24 [150/20] via 172.1.1.2, 00:49:57, gigabitethernet0
OE  120.1.1.0/24 [150/20] via 172.1.1.2, 02:22:08, gigabitethernet0
OE  171.1.1.0/24 [150/20] via 172.1.1.2, 02:22:41, gigabitethernet0
```

According to the route table of Device1, the RIP routes 100.1.1.0/24, 110.1.1.0/24, and 120.1.1.0/24 are redistributed to the OSPF process and successfully advertised to Device1.

Step 5: Configure an ACL and routing policy.

#Configure Device2.

Configure an ACL to allow routes 100.1.1.0/24, 110.1.1.0/24, and 120.1.1.0/24 to pass.

```
Device2(config)#ip access-list standard 1
Device2(config-std-nacl)#permit 100.1.1.0 0.0.0.255
Device2(config-std-nacl)#exit
Device2(config)#ip access-list standard 2
Device2(config-std-nacl)#permit 110.1.1.0 0.0.0.255
Device2(config-std-nacl)#exit
Device2(config)#ip access-list standard 3
Device2(config-std-nacl)#permit 120.1.1.0 0.0.0.255
Device2(config-std-nacl)#exit
```

Configure routing policy rip_to_ospf. Set the tag property of the routes that match ACL 1, set the metric property of the routes that match ACL2, and do not change the routing properties of the routes that match ACL 3.

```
Device2(config)#route-map rip_to_ospf 10
Device2(config-route-map)#match ip address 1
Device2(config-route-map)#set tag 5
Device2(config-route-map)#exit
Device2(config)#route-map rip_to_ospf 20
Device2(config-route-map)#match ip address 2
Device2(config-route-map)#set metric 50
Device2(config-route-map)#exit
Device2(config)#route-map rip_to_ospf 30
Device2(config-route-map)#match ip address 3
Device2(config-route-map)#exit
```

Note:

- In configuring a routing policy, you can create a matching rule based on a prefix list or ACL. The prefix list can precisely match routing masks while the ACL cannot match routing masks.

Step 6: Configure OSPF to redistribute RIP routes and associate a routing policy.

#Configure Device2.

```
Device2(config)#router ospf 100
Device2(config-ospf)#redistribute rip route-map rip_to_ospf
Device2(config-ospf)#exit
```

Step 7: Check the result.

#Check the OSPF database of Device1.

```
Device1#show ip ospf database external
      OSPF Router with ID (172.1.1.1) (Process ID 100)


        AS External Link States


  LS age: 1183
 Options: 0x22 (-|-|DC|-|-|-|E|-)
 LS Type: AS-external-LSA
 Link State ID: 100.1.1.0 (External Network Number)
 Advertising Router: 172.1.1.2
 LS Seq Number: 80000006
 Checksum: 0xbcc0
 Length: 36
 Network Mask: /24
    Metric Type: 2 (Larger than any link state path)
    TOS: 0
    Metric: 20
    Forward Address: 0.0.0.0
    External Route Tag: 5

 LS age: 1233
 Options: 0x22 (-|-|DC|-|-|-|E|-)
 LS Type: AS-external-LSA
 Link State ID: 110.1.1.0 (External Network Number)
 Advertising Router: 172.1.1.2
 LS Seq Number: 80000006
```

```
        Checksum: 0x0d4d
        Length: 36
        Network Mask: /24
              Metric Type: 2 (Larger than any link state path)
              TOS: 0
              Metric: 50
              Forward Address: 0.0.0.0
              External Route Tag: 0

        LS age: 1113
        Options: 0x22 (-|-|DC|-|-|-|E|-)
        LS Type: AS-external-LSA
        Link State ID: 120.1.1.0 (External Network Number)
        Advertising Router: 172.1.1.2
        LS Seq Number: 80000005
        Checksum: 0x5f10
        Length: 36
        Network Mask: /24
              Metric Type: 2 (Larger than any link state path)
              TOS: 0
              Metric: 20
              Forward Address: 0.0.0.0
              External Route Tag: 0
      #Query the route table of Device1.
      Device1#show ip route
      Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
             U - Per-user Static route
             O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

      OE  100.1.1.0/24 [150/20] via 172.1.1.2, 02:30:28, gigabitethernet0
      OE  110.1.1.0/24 [150/50] via 172.1.1.2, 00:58:17, gigabitethernet0
      OE  120.1.1.0/24 [150/20] via 172.1.1.2, 02:30:28, gigabitethernet0
```

According to the OSPF database and route table of Device1, the tag of route 100.1.1.0/24 is 5, the metric of route 110.1.1.0/24 is 50, and the routing properties of route 120.1.1.0/24 are not changed.

Note:

- In redistributing external routes, the routes of the direct connect interfaces that are covered by the RIP process will also be redistributed into the target protocol.

15.3.2. Configure Routing Policy for BGP

Network Requirements

- Run IGP protocol ISPF and set up IBGP neighbors between Device1 and Device2 and between Device1 and Device3, and set up EBGP neighbors between Device4 and Device2 and between Device4 and Device3.
- Configure a routing policy on Device2 and Device3 so that the data of Device1 reaches network segment 100.1.1.0/24 through Device2, reaches network segment 110.1.1.0/24 through Device3, reaches network segment 120.1.1.0/24 through Device2, and reaches network segment 130.1.1.0/24 through Device3.

Network Topology

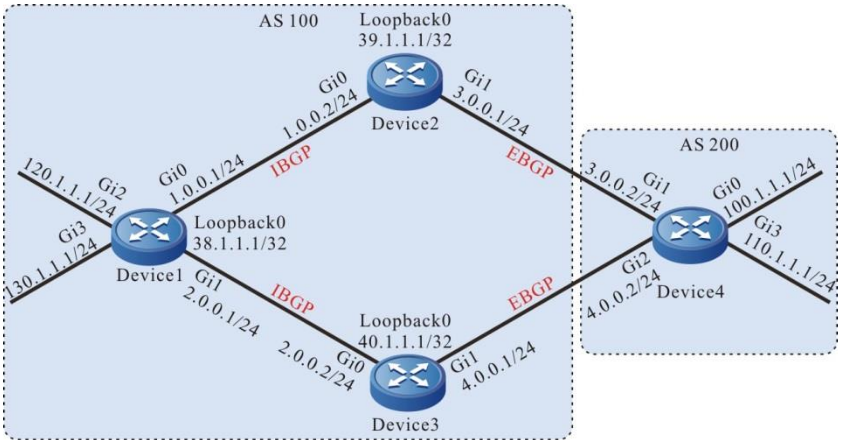

Figure 15–2 Configuring a routing policy for BGP

Configuration Steps

Step 1: Configure the IP addresses of the interfaces. (Omitted)

Step 2: Configure OSPF so that loopback routes are reachable between devices.

#Configure Device1.

```
Device1#configure terminal
Device1(config)#router ospf 100
Device1(config-ospf)#network 1.0.0.0 0.0.0.255 area 0
Device1(config-ospf)#network 2.0.0.0 0.0.0.255 area 0
Device1(config-ospf)#network 38.1.1.1 0.0.0.0 area 0
Device1(config-ospf)#exit
```

#Configure Device2.

```
Device2#configure terminal
Device2(config)#router ospf 100
Device2(config-ospf)#network 1.0.0.0 0.0.0.255 area 0
Device2(config-ospf)#network 39.1.1.1 0.0.0.0 area 0
Device2(config-ospf)#exit
```

#Configure Device3.

```
Device3#configure terminal
```

```
Device3(config)#router ospf 100
Device3(config-ospf)#network 2.0.0.0 0.0.0.255 area 0
Device3(config-ospf)#network 40.1.1.1 0.0.0.0 area 0
Device3(config-ospf)#exit
```

#Query the route table of Device1.

```
Device1#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
     U - Per-user Static route
     O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

O   39.1.1.1/32 [110/2] via 1.0.0.2, 19:11:33, gigabitethernet0
O   40.1.1.1/32 [110/2] via 2.0.0.2, 18:56:32, gigabitethernet1
```

#Query the route table of Device2.

```
Device2#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
     U - Per-user Static route
     O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

O   2.0.0.0/24 [110/2] via 1.0.0.1, 19:19:10, gigabitethernet0
O   38.1.1.1/32 [110/2] via 1.0.0.1, 19:09:43, gigabitethernet0
O   40.1.1.1/32 [110/3] via 1.0.0.1, 18:56:49, gigabitethernet0
```

#Query the route table of Device3.

```
Device3#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
     U - Per-user Static route
     O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

O   1.0.0.0/24 [110/2] via 2.0.0.1, 19:17:33, gigabitethernet0
O   38.1.1.1/32 [110/2] via 2.0.0.1, 19:09:59, gigabitethernet0
O   39.1.1.1/32 [110/3] via 2.0.0.1, 19:12:06, gigabitethernet0
```

After the configuration is completed, Device1 can set up OSPF neighbors respectively with Device2 and Device3 and the devices can learn the Loopback routes of the peer end.

Step 3: Configure BGP.

#Configure Device1.

Configure Device1 to set up IBGP neighbors respectively with Device2 and Device3 through Loopback interfaces and advertises routes 120.1.1.0/24 and 130.1.1.0/24 to the BGP route table.

```
Device1(config)#router bgp 100
```

```
Device1(config-bgp)#neighbor 39.1.1.1 remote-as 100
Device1(config-bgp)#neighbor 39.1.1.1 update-source loopback0
Device1(config-bgp)#neighbor 40.1.1.1 remote-as 100
Device1(config-bgp)#neighbor 40.1.1.1 update-source loopback0
Device1(config-bgp)#network 120.1.1.0 255.255.255.0
Device1(config-bgp)#network 130.1.1.0 255.255.255.0
Device1(config-bgp)#exit
```

#Configure Device2.

```
Device2(config)#router bgp 100
Device2(config-bgp)#neighbor 38.1.1.1 remote-as 100
Device2(config-bgp)#neighbor 38.1.1.1 update-source loopback0
Device2(config-bgp)#neighbor 38.1.1.1 next-hop-self
Device2(config-bgp)#neighbor 3.0.0.2 remote-as 200
Device2(config-bgp)#exit
```

#Configure Device3.

```
Device3(config)#router bgp 100
Device3(config-bgp)#neighbor 38.1.1.1 remote-as 100
Device3(config-bgp)#neighbor 38.1.1.1 update-source loopback0
Device3(config-bgp)#neighbor 38.1.1.1 next-hop-self
Device3(config-bgp)#neighbor 4.0.0.2 remote-as 200
Device3(config-bgp)#exit
```

#Configure Device4.

Configure Device4 to set up EBGP neighbors respectively with Device2 and Device3 and advertise routes 100.1.1.0/24 and 110.1.1.0/24 to the BGP route table.

```
Device4#configure terminal
Device4(config)#router bgp 200
Device4(config-bgp)#neighbor 3.0.0.1 remote-as 100
Device4(config-bgp)#neighbor 4.0.0.1 remote-as 100
Device4(config-bgp)#network 100.1.1.0 255.255.255.0
Device4(config-bgp)#network 110.1.1.0 255.255.255.0
Device4(config-bgp)#exit
```

#Query the BGP routing information of Device1.

```
Device1#show ip bgp
BGP table version is 2, local router ID is 38.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
        S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
```

```
    Network          Next Hop         Metric LocPrf Weight Path
[B]*>i100.1.1.0/24       39.1.1.1           0    100      0 200 i
[B]* i                40.1.1.1           0    100      0 200 i
[B]*>i110.1.1.0/24       39.1.1.1           0    100      0 200 i
[B]* i                40.1.1.1           0    100      0 200 i
[B]*> 120.1.1.0/24       0.0.0.0              0        32768 i
[B]*> 130.1.1.0/24       0.0.0.0              0        32768 i
```

#Query the route table of Device1.

```
Device1#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
     U - Per-user Static route
     O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external


B   100.1.1.0/24 [200/0] via 39.1.1.1, 19:03:19, gigabitethernet0
B   110.1.1.0/24 [200/0] via 39.1.1.1, 19:03:19, gigabitethernet0
```

According to the BGP route table of Device1, data that are targeted at network segments 100.1.1.0/24 and 110.1.1.0/24 have two valid routes respectively. Because the router ID of Device2 is smaller, so the BGP data that are targeted at network segments 100.1.1.0/24 and 110.1.1.0/24 choose to pass Device2 by default.

#Query the BGP routing information of Device4.

```
Device4#show ip bgp
BGP table version is 3, local router ID is 110.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
           S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
    Network          Next Hop         Metric LocPrf Weight Path
[B]*> 100.1.1.0/24       0.0.0.0              0        32768 i
[B]*> 110.1.1.0/24       0.0.0.0              0        32768 i
[B]*  120.1.1.0/24       4.0.0.1              0            0 100 i
[B]*>                 3.0.0.1              0            0 100 i
[B]*  130.1.1.0/24       4.0.0.1              0            0 100 i
[B]*>                 3.0.0.1              0            0 100 i
```

#Query the route table of Device4.

```
Device4#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
     U - Per-user Static route
     O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external
```

```
B   120.1.1.0/24 [20/0] via 3.0.0.1, 19:25:05, gigabitethernet1
B   130.1.1.0/24 [20/0] via 3.0.0.1, 19:25:05, gigabitethernet1
```

According to the BGP route table of Device4, the data that are targeted at network segments 120.1.1.0/24 and 130.1.1.0/24 have two valid routes. Because Device4 first sets up a neighbor relation with Device2, it takes longer time for Device2 to learn the two routes, so BGP data that are targeted at the network segments 120.1.1.0/24 and 130.1.1.0/24 choose to pass Device2 by default.

Step 4: Configure a prefix list and routing policy.

#Configure Device2.

Configure a prefix list to allow routes 100.1.1.0/24 and 130.1.1.0/24 to pass.

```
Device2(config)#ip prefix-list 1 permit 100.1.1.0/24
Device2(config)#ip prefix-list 2 permit 130.1.1.0/24
```

Configure the routing policy lp so that the prefix list 1 of Device2 allows setting local-preference for routes.

```
Device2(config)#route-map lp 10
Device2(config-route-map)#match ip address prefix-list 1
Device2(config-route-map)#set local-preference 200
Device2(config-route-map)#exit
Device2(config)#route-map lp 20
Device2(config-route-map)#exit
```

Configure the routing policy med so that the prefix list 2 of Device2 allows setting the MED property for routes.

```
Device2(config)#route-map med 10
Device2(config-route-map)#match ip address prefix-list 2
Device2(config-route-map)#set metric 10
Device2(config-route-map)#exit
Device2(config)#route-map med 20
Device2(config-route-map)#exit
```

#Configure Device3.

Configure a prefix list to allow routes 110.1.1.0/24 and 120.1.1.0/24 to pass.

```
Device3(config)#ip prefix-list 1 permit 110.1.1.0/24
Device3(config)#ip prefix-list 2 permit 120.1.1.0/24
```

Configure the routing policy lp so that the prefix list 1 of Device3 allows setting local-preference for routes.

```
Device3(config)#route-map lp 10
Device3(config-route-map)#match ip address prefix-list 1
Device3(config-route-map)#set local-preference 200
Device3(config-route-map)#exit
```

```
Device3(config)#route-map lp 20
Device3(config-route-map)#exit
```

Configure the routing policy med so that the prefix list 2 of Device3 allows setting the MED property for routes.

```
Device3(config)#route-map med 10
Device3(config-route-map)# match ip address prefix-list 2
Device3(config-route-map)#set metric 10
Device3(config-route-map)#exit
Device3(config)#route-map med 20
Device3(config-route-map)#exit
```

Note:

- In configuring a routing policy, you can create a matching rule based on a prefix list or ACL. The prefix list can precisely match routing masks while the ACL cannot match routing masks.

Step 5: Configure a routing policy for BGP.

#Configure Device2.

Apply the routing policy lp to the outgoing routes of neighbor 38.1.1.1 and apply the routing policy med to the outgoing routes of neighbor 3.0.0.2.

```
Device2(config)#router bgp 100
Device2(config-bgp)#neighbor 38.1.1.1 route-map lp out
Device2(config-bgp)#neighbor 3.0.0.2 route-map med out
Device2(config-bgp)#exit
```

#Configure Device3.

Apply the routing policy lp to the outgoing routes of neighbor 38.1.1.1 and apply the routing policy med to the outgoing routes of neighbor 4.0.0.2.

```
Device3(config)#router bgp 100
Device3(config-bgp)#neighbor 38.1.1.1 route-map lp out
Device3(config-bgp)#neighbor 4.0.0.2 route-map med out
Device3(config-bgp)#exit
```

Step 6: Check the result.

#Query the BGP routing information of Device1.

```
Device1#show ip bgp
BGP table version is 9, local router ID is 38.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
          S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
    Network          Next Hop          Metric LocPrf Weight Path
```

```
[B]* i100.1.1.0/24      40.1.1.1            0    100      0 200 i
[B]*>i                39.1.1.1            0    200      0 200 i
[B]*>i110.1.1.0/24      40.1.1.1            0    200      0 200 i
[B]* i                39.1.1.1            0    100      0 200 i
[B]*> 120.1.1.0/24      0.0.0.0              0        32768 i
[B]*> 130.1.1.0/24      0.0.0.0              0        32768 i
```

#Query the route table of Device1.

```
Device1#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external

B   100.1.1.0/24 [200/0] via 39.1.1.1, 02:58:12, gigabitethernet0
B   110.1.1.0/24 [200/0] via 40.1.1.1, 02:58:10, gigabitethernet1
```

According to the BGP route table of Device1, route 100.1.1.0/24 has two next hops, 40.1.1.1 and 39.1.1.1. The local-preference of the route with the next hop 39.1.1.1 has been changed to 200 so that the data that are targeted at the network segment 100.1.1.0/24 choose to pass Device2 with priority. Route 110.1.1.0/24 also has two next hops, 40.1.1.1 and 39.1.1.1. The local-preference of the route with the next hop 40.1.1.1 has been changed to 200 so that the data that are targeted at the network segment 110.1.1.0/24 choose to pass Device3 with priority.

#Query the BGP routing information of Device4.

```
Device4#show ip bgp
BGP table version is 9, local router ID is 110.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
          S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
    Network          Next Hop          Metric LocPrf Weight Path
[B]*> 100.1.1.0/24      0.0.0.0              0        32768 i
[B]*> 110.1.1.0/24      0.0.0.0              0        32768 i
[B]*  120.1.1.0/24      4.0.0.1             10           0 100 i
[B]*>                 3.0.0.1            0           0 100 i
[B]*> 130.1.1.0/24      4.0.0.1              0           0 100 i
[B]*                  3.0.0.1           10           0 100 i
```

#Query the route table of Device4.

```
Device4#show ip route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management, E - IRMP, EX - IRMP external
```

```
B   120.1.1.0/24 [20/0] via 3.0.0.1, 03:05:39, gigabitethernet1
B   130.1.1.0/24 [20/0] via 4.0.0.1, 03:05:37, gigabitethernet2
```

According to the BGP route table of Device4, route 120.1.1.0/24 has two next hops, 4.0.0.1 and 3.0.0.1. The metric of the route with the next hop 4.0.0.1 has been changed to 10 so that the data that are targeted at the network segment 120.1.1.0/24 choose to pass Device2 with priority. Route 130.1.1.0/24 also has two next hops, 4.0.0.1 and 3.0.0.1. The metric of the route with the next hop 3.0.0.1 has been changed to 10 so that the data that are targeted at the network segment 130.1.1.0/24 choose to pass Device3 with priority.

Note:

- If a routing policy is applied to a BGP peer or peer group, it can be applied in the receiving or advertisement direction of the BGP peer or peer group, and the settings take effect after BGP is reset.

15.3.3. Configure IPv6 Route Redistribution with the Routing Policy

Network Requirements

- Run OSPFv3 between Device1 and Device2, and run RIPng between Device2 and Device3.
- On Device2, configure OSPFv3 to redistribute RIPng routes, and associate a routing policy to modify route properties. It is required that the tag property of route 2005:1::1/64 is changed to 5, the metric value of route 2006:1::1/64 is changed to 50, and the property of route 2007:1::1/64 keeps unchanged.

Network Topology

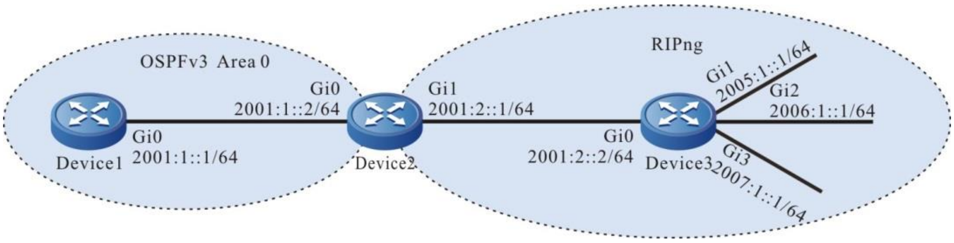

Figure 15-3 Networking for configuring the IPv6 route re-distribution and associate the routing policy

Configuration Steps

Step 1: Configure the IPv6 addresses of the interfaces. (Omitted)

Step 2: Configure OSPFv3.

#Configure Device1, configure an OSPFv3 process, and cover the interface to area 0.

```
Device1#configure terminal
Device1(config)#ipv6 router ospf 1
Device1(config-ospf6)#router-id 1.1.1.1
Device1(config-ospf6)#exit
Device1(config)#interface gigabitethernet0
Device1(config-if-gigabitethernet0)#ipv6 router ospf 1 area 0
Device1(config-if-gigabitethernet0)#exit
```

#Configure Device2, configure an OSPFv3 process, and cover the interface to area 0.

```
Device2#configure terminal
Device2(config)#ipv6 router ospf 1
Device2(config-ospf6)#router-id 2.2.2.2
Device2(config-ospf6)#exit
Device2(config)#interface gigabitethernet0
Device2(config-if-gigabitethernet0)#ipv6 router ospf 1 area 0
Device2(config-if-gigabitethernet0)#exit
```

Step 3: Configure RIP.

#Configure Device2, configure a RIPng process, and cover the interface to RIPng.

```
Device2(config)#ipv6 router rip 1
Device2(config-ripng)#exit
Device2(config)#interface gigabitethernet1
Device2(config-if-gigabitethernet1)#ipv6 rip enable 1
Device2(config-if-gigabitethernet1)#exit
```

#Configure Device3, configure a RIPng process, and cover the interface to RIPng.

```
Device3#configure terminal
Device3(config)#ipv6 router rip 1
Device3(config-ripng)#exit
Device3(config)#interface gigabitethernet0
Device3(config-if-gigabitethernet0)#ipv6 rip enable 1
Device3(config-if-gigabitethernet0)#exit
Device3(config)#interface gigabitethernet1
Device3(config-if-gigabitethernet1)#ipv6 rip enable 1
Device3(config-if-gigabitethernet1)#exit
Device3(config)#interface gigabitethernet2
Device3(config-if-gigabitethernet2)#ipv6 rip enable 1
Device3(config-if-gigabitethernet2)#exit
Device3(config)#interface gigabitethernet0/2/3
Device3(config-if-gigabitethernet0/2/3)#ipv6 rip enable 1
Device3(config-if-gigabitethernet0/2/3)#exit
```

Step 4: Configure OSPFv3 to re-distribute the RIPng route.

#Configure Device2.

```
Device2(config)#ipv6 router ospf 1
Device2(config-ospf6)#redistribute rip 1
Device2(config-ospf6)#exit
```

#View the IPv6 core route table of Device1.

```
Device1#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management
C   2001:1::/64 [0/0]
    via ::, 00:01:10, gigabitethernet0
L   2001:1::1/128 [0/0]
    via ::, 00:01:10, gigabitethernet0
OE  2001:2::/64 [150/20]
    via fe80::201:7aff:fedf:e4d5, 00:00:14, gigabitethernet0
OE  2005:1::/64 [150/20]
    via fe80::201:7aff:fedf:e4d5, 00:00:14, gigabitethernet0
OE  2006:1::/64 [150/20]
    via fe80::201:7aff:fedf:e4d5, 00:00:14, gigabitethernet0
OE  2007:1::/64 [150/20]
    via fe80::201:7aff:fedf:e4d5, 00:00:14, gigabitethernet0
```

According to the IPv6 core route table of Device1, the RIPng route 2005:1::1/64, 2006:1::1/64, and 2007:1::1/64 on Device2 are re-distributed to the OSPFv3 process, and successfully advertised to Device1.

Step 5: Configure the IPv6 prefix list and routing policy.

#Configure Device2.

Configure the IPv6 prefix list, permitting 2005:1::1/64, 2006:1::1/64 and 2007:1::1/64 routes to pass.

```
Device2(config)#ipv6 prefix-list 7001 permit 2005:1::/64
Device2(config)#ipv6 prefix-list 7002 permit 2006:1::/64
Device2(config)#ipv6 prefix-list 7003 permit 2007:1::/64
```

Configure routing policy ripng_to_ospfv3. Set the tag property of the routes that match ACL 7001, set the metric property of the routes that match ACL7002, and do not change the routing properties of the routes that match ACL 7003.

```
Device2(config)#route-map ripng_to_ospfv3 10
Device2(config-route-map)#match ipv6 address prefix-list 7001
Device2(config-route-map)#set tag 5
```

```
Device2(config-route-map)#exit
Device2(config)#route-map ripng_to_ospfv3 20
Device2(config-route-map)#match ipv6 address prefix-list 7002
Device2(config-route-map)#set metric 50
Device2(config-route-map)#exit
Device2(config)#route-map ripng_to_ospfv3 30
Device2(config-route-map)#match ipv6 address prefix-list 7003
 Device2(config-route-map)#exit
```

Note:

- When configuring the routing policy, both the prefix list and ACL can create the match rule, and the difference is that the prefix list can correctly match the route mask, but the ACL cannot match the route mask.

Step 6: Configure OSPFv3 to re-distribute the RIPng route with the Routing Policy.

#Configure Device2.

```
Device2(config)#ipv6 router ospf 1
Device2(config-ospf6)#redistribute rip 1 route-map ripng_to_ospfv3
Device2(config-ospf6)#exit
```

Step 7: Check the result.

#View the OSPFv3 database of Device1.

```
Device1#show ipv6 ospf 1 database external

            OSPFv3 Router with ID (1.1.1.1) (Process 1)

                AS-external-LSA
  LS age: 110
  LS Type: AS-External-LSA
  Link State ID: 0.0.0.2
  Advertising Router: 2.2.2.2
  LS Seq Number: 0x80000002
  Checksum: 0xD2AE
  Length: 40
    Metric Type: 2 (Larger than any link state path)
    Metric: 20
    Prefix: 2005:1::/64
    Prefix Options: 0 (-|-|-|-|-)
    External Route Tag: 5
  LS age: 110
```

```
 LS Type: AS-External-LSA
 Link State ID: 0.0.0.3
 Advertising Router: 2.2.2.2
 LS Seq Number: 0x80000002
 Checksum: 0x105B
 Length: 36
   Metric Type: 2 (Larger than any link state path)
   Metric: 50
   Prefix: 2006:1::/64
   Prefix Options: 0 (-|-|-|-|-)
 LS age: 1066
 LS Type: AS-External-LSA
 Link State ID: 0.0.0.4
 Advertising Router: 2.2.2.2
 LS Seq Number: 0x80000001
 Checksum: 0x5F29
 Length: 36
   Metric Type: 2 (Larger than any link state path)
   Metric: 20
   Prefix: 2007:1::/64
   Prefix Options: 0 (-|-|-|-|-)
```

#View the IPv6 core route table of Device1.

```
Device1#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
       U - Per-user Static route
       O - OSPF, OE-OSPF External, M - Management
C   2001:1::/64 [0/0]
     via ::, 00:18:19, gigabitethernet0
L   2001:1::1/128 [0/0]
     via ::, 00:18:19, gigabitethernet0
OE  2005:1::/64 [150/20]
     via fe80::201:7aff:fedf:e4d5, 00:17:24, gigabitethernet0
OE  2006:1::/64 [150/50]
     via fe80::201:7aff:fedf:e4d5, 00:17:24, gigabitethernet0
OE  2007:1::/64 [150/20]
     via fe80::201:7aff:fedf:e4d5, 00:17:24, gigabitethernet0
```

According to the OSPFv3 database and route table of Device1, the route Tag of 2005:1::/64 is 5, the route Metric of 2006:1::/64, and the route property of 2007:1::/64 is not changed.

Note:

- When re-distributing the external route, the direct-connect interface route covered by the RIPng process is also re-distributed to the target protocol.

15.3.4. Configure IPv6 BGP with Routing Policy

Network Requirements

- Run the IGP OSPFv3 between Device1 and Device2, Device3 respectively, and set up the EBGP neighbor between Device4 and Device2, Device3 respectively.
- It is required to configure the routing policy on Device2 and Device3 so that Device1 forwards the data to 2001:7::/64 via Device3 and the data to 2001:8::/64 via Device2, and Device4 forwards the data to 2001:1::/64 via Device3, and the data to 2001:2::/64 via Device2.

Network Topology

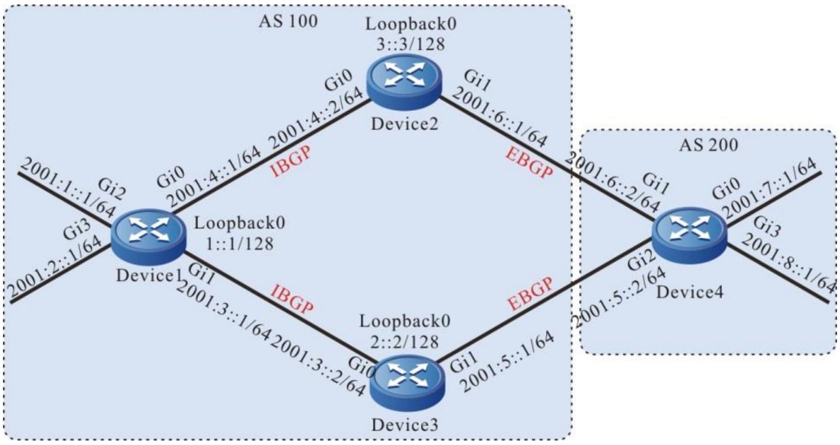

Figure 15-4 Networking for configuringIPv6 BGP with the routing policy

Configuration Steps

Step 1: Configure the IP addresses of the interfaces. (Omitted)

Step 2: Configure OSPFv3 so that loopback routes are reachable between devices.

#Configure Device1.

```
Device1#configure terminal
Device1(config)#ipv6 router ospf 1
Device1(config-ospf6)#router-id 1.1.1.1
Device1(config-ospf6)#exit
Device1(config)#interface gigabitethernet0
Device1(config-if-gigabitethernet0)#ipv6 router ospf 1 area 0
Device1(config-if-gigabitethernet0)#exit
Device1(config)#interface gigabitethernet1
Device1(config-if-gigabitethernet1)#ipv6 router ospf 1 area 0
Device1(config-if-gigabitethernet1)#exit
```

```
Device1(config)#interface loopback 0
Device1(config-if-loopback0)#ipv6 router ospf 1 area 0
Device1(config-if-loopback0)#exit
```

#Configure Device2.

```
Device2#configure terminal
Device2(config)#ipv6 router ospf 1
Device2(config-ospf6)#router-id 2.2.2.2
Device2(config-ospf6)#exit
Device2(config)#interface gigabitethernet0
Device2(config-if-gigabitethernet0)#ipv6 router ospf 1 area 0
Device2(config-if-gigabitethernet0)#exit
Device2(config)# interface loopback 0
Device2(config-if-loopback0)#ipv6 router ospf 1 area 0
Device2(config-if-loopback0)#exit
```

#Configure Device3.

```
Device3#configure terminal
Device3(config)#ipv6 router ospf 1
Device3(config-ospf6)#router-id 3.3.3.3
Device3(config-ospf6)#exit
Device3(config)#interface gigabitethernet0
Device3(config-if-gigabitethernet0)#ipv6 router ospf 1 area 0
Device3(config-if-gigabitethernet0)#exit
Device3(config)# interface loopback 0
Device3(config-if-loopback0)#ipv6 router ospf 1 area 0
Device3(config-if-loopback0)#exit
```

#View the IPv6 core route table of Device1.

```
Device1#show ipv6 route ospf
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
     U - Per-user Static route
     O - OSPF, OE-OSPF External, M - Management

O   2::2/128 [110/1]
    via fe80::201:7aff:fedf:e4d5, 00:29:32, gigabitethernet0
O   3::3/128 [110/1]
    via fe80::201:7aff:fec9:1ca2, 00:28:57, gigabitethernet1
```

#View the IPv6 core route table of Device2.

```
Device2#show ipv6 route ospf
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
     U - Per-user Static route
     O - OSPF, OE-OSPF External, M - Management

O   1::1/128 [110/1]
    via fe80::201:7aff:fe46:af72, 00:38:27, gigabitethernet0
O   3::3/128 [110/2]
    via fe80::201:7aff:fe46:af72, 00:36:37, gigabitethernet0
```

#View the IPv6 core route table of Device3.

```
Device3#show ipv6 route ospf
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
     U - Per-user Static route
     O - OSPF, OE-OSPF External, M - Management

O   1::1/128 [110/1]
    via fe80::201:7aff:fe46:af77, 00:40:25, gigabitethernet0
O   2::2/128 [110/2]
    via fe80::201:7aff:fe46:af77, 00:39:10, gigabitethernet0
```

After configuration, Device1 can set up the OSPFv3 neighbor with Device2, Device3 respectively, and learn the peer Loopback route mutually.

Step 3: Configure IPv6 BGP.

#Configure Device1.

Configure Device1 to use the Loopback interface address to set up the IBGP neighbor with Device2, Device3 respectively, and advertise the route 2001:1::1/64, 2001:2::1/64 to the IPv6 BGP route table.

```
Device1(config)#router bgp 100
Device1(config-bgp)#bgp router-id 1.1.1.1
Device1(config-bgp)#neighbor 2::2 remote-as 100
Device1(config-bgp)#neighbor 2::2 update-source loopback 0
Device1(config-bgp)#neighbor 3::3 remote-as 100
Device1(config-bgp)#neighbor 3::3 update-source loopback 0
Device1(config-bgp)#address-family ipv6 unicast
Device1(config-bgp-af)#neighbor 2::2 activate
```

```
Device1(config-bgp-af)#neighbor 3::3 activate
Device1(config-bgp-af)#network 2001:1::/64
Device1(config-bgp-af)#network 2001:2::/64
Device1(config-bgp-af)#exit-address-family
Device1(config-bgp)#exit
```

#Configure Device2.

```
Device2(config)#router bgp 100
Device2(config-bgp)#bgp router-id 2.2.2.2
Device2(config-bgp)#neighbor 1::1 remote-as 100
Device2(config-bgp)#neighbor 1::1 update-source loopback 0
Device2(config-bgp)#neighbor 2001:5::2 remote-as 200
Device2(config-bgp)#address-family ipv6 unicast
Device2(config-bgp-af)#neighbor 1::1 activate
Device2(config-bgp-af)#neighbor 2001:5::2 activate
Device2(config-bgp-af)#neighbor 1::1 next-hop-self
Device2(config-bgp-af)#exit-address-family
Device2(config-bgp)#exit
```

#Configure Device3.

```
Device3(config)#router bgp 100
Device3(config-bgp)#bgp router-id 3.3.3.3
Device3(config-bgp)#neighbor 1::1 remote-as 100
Device3(config-bgp)#neighbor 1::1 update-source loopback 0
Device3(config-bgp)#neighbor 2001:6::2 remote-as 200
Device3(config-bgp)#address-family ipv6 unicast
Device3(config-bgp-af)#neighbor 1::1 activate
Device3(config-bgp-af)#neighbor 2001:6::2 activate
Device3(config-bgp-af)#neighbor 1::1 next-hop-self
Device3(config-bgp-af)#exit-address-family
Device3(config-bgp)#exit
```

#Configure Device4.

Configure Device4 to use the Loopback interface address to set up the EBGP neighbor with Device2, Device3 respectively, and advertise the route 2001 :7 ::/64, 2001 :8 ::/64 to the BGP route table.

```
Device4(config)#router bgp 200
Device4(config-bgp)#bgp router-id 4.4.4.4
Device4(config-bgp)#neighbor 2001:5::1 remote-as 100
Device4(config-bgp)#neighbor 2001:6::1 remote-as 100
```

```
Device4(config-bgp)#address-family ipv6 unicast
Device4(config-bgp-af)#neighbor 2001:5::1 activate
Device4(config-bgp-af)#neighbor 2001:6::1 activate
Device4(config-bgp-af)#network 2001:7::/64
Device4(config-bgp-af)#network 2001:8::/64
Device4(config-bgp-af)#exit-address-family
Device4(config-bgp)#exit
```

#View the BGP route information of Device1.

```
Device1#show bgp ipv6 unicast
BGP table version is 4, local router ID is 1.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
          S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
    Network           Next Hop          Metric    LocPrf Weight Path
[B]*> 2001:1::/64       ::                 0          32768 i
[B]*> 2001:2::/64       ::                 0          32768 i
[B]* i2001:7::/64       3::3               0       100      0 200 i
[B]*>i                2::2               0       100      0 200 i
[B]* i2001:8::/64       3::3               0       100      0 200 i
[B]*>i                2::2               0       100      0 200 i
```

#View the IPv6 core route table of Device1.

```
Device1#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
     U - Per-user Static route
     O - OSPF, OE-OSPF External, M - Management

O   2::2/128 [110/1]
    via fe80::201:7aff:fedf:e4d5, 17:56:30, gigabitethernet0
O   3::3/128 [110/1]
    via fe80::201:7aff:fec9:1ca2, 19:14:42, gigabitethernet1
C   2001:1::/64 [0/0]
    via ::, 19:48:30, gigabitethernet0/2/3
L   2001:1::1/128 [0/0]
    via ::, 19:48:30, gigabitethernet0/2/3
C   2001:2::/64 [0/0]
    via ::, 00:19:44, gigabitethernet2
```

```
L   2001:2::1/128 [0/0]
    via ::, 00:19:44, gigabitethernet2
C   2001:3::/64 [0/0]
    via ::, 18:11:16, gigabitethernet0
L   2001:3::1/128 [0/0]
    via ::, 18:11:16, gigabitethernet0
C   2001:4::/64 [0/0]
    via ::, 19:45:33, gigabitethernet1
L   2001:4::1/128 [0/0]
    via ::, 19:45:33, gigabitethernet1
B   2001:7::/64 [200/0]
    via 2::2, 00:03:29, gigabitethernet0
B   2001:8::/64 [200/0]
    via 2::2, 00:03:29, gigabitethernet0
```

According to the IPv6 BGP route table of Device1, the data to network segments 2001:7::/64 and 2001:8::/64 have two valid routes. The router ID of Device2 is small, so BGP data to the network segments 2001:7::/64 and 2001:8::/64 chooses to pass Device2 by default.

#View the IPv6 BGP route information of Device4.

```
Device4#show bgp ipv6 unicast
BGP table version is 4, local router ID is 4.4.4.4
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
           S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
    Network          Next Hop         Metric    LocPrf Weight Path
[B]*  2001:1::/64       2001:6::1          0             0 100 i
[B]*>                2001:5::1          0             0 100 i
[B]*  2001:2::/64       2001:6::1          0             0 100 i
[B]*>                2001:5::1          0             0 100 i
[B]*> 2001:7::/64       ::                 0          32768 i
[B]*> 2001:8::/64       ::                 0          32768 i
```

#View the IPv6 core route table of Device4.

```
Device4#show ipv6 route
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
     U - Per-user Static route
     O - OSPF, OE-OSPF External, M - Management

B   2001:1::/64 [20/0]
```

```
    via 2001:5::1, 00:10:52, gigabitethernet0
B   2001:2::/64 [20/0]
    via 2001:5::1, 00:10:52, gigabitethernet0
C   2001:5::/64 [0/0]
    via ::, 19:48:03, gigabitethernet0
L   2001:5::2/128 [0/0]
    via ::, 19:48:03, gigabitethernet0
C   2001:6::/64 [0/0]
    via ::, 19:48:35, gigabitethernet1
L   2001:6::2/128 [0/0]
    via ::, 19:48:35, gigabitethernet1
C   2001:7::/64 [0/0]
    via ::, 19:45:20, gigabitethernet0/2/3
L   2001:7::1/128 [0/0]
    via ::, 19:45:20, gigabitethernet0/2/3
C   2001:8::/64 [0/0]
    via ::, 19:45:08, gigabitethernet2
L   2001:8::1/128 [0/0]
    via ::, 19:45:08, gigabitethernet2
```

According to the BGP route table of Device4, the data to network segments 2001:1::/64 and 2001:2::/64 have two valid routes. Because Device4 first sets up a neighbor with Device2, it takes longer time for Device2 to learn the two routes, so the BGP data to network segments 2001:1::/64 and 2001:2::/64 choose to pass Device2 by default.

Step 4: Configure the prefix list and routing policy.

#Configure Device2.

Configure the prefix list, permitting 2001:8::/64, 2001:1::/64 routes to pass.

```
Device2(config)#ipv6 prefix-list 7001 permit 2001:8::/64
Device2(config)#ipv6 prefix-list 7002 permit 2001:1::/64
```

Configure the routing policy lp so that Device2 allows setting local-preference for routes that match the prefix list 7001.

```
Device2(config)#route-map lp 10
Device2(config-route-map)#match ipv6 address 7001
Device2(config-route-map)#set local-preference 200
Device2(config-route-map)#exit
Device2(config)#route-map lp 20
Device2(config-route-map)#exit
```

Configure the routing policy med so that Device2 allows setting the MED attribute for routes that match the prefix list 7002.

```
Device2(config)#route-map med 10
Device2(config-route-map)#match ipv6 address 7002
Device2(config-route-map)#set metric 10
Device2(config-route-map)#exit
Device2(config)#route-map med 20
Device2(config-route-map)#exit
```

#Configure Device3.

Configure the prefix list, permitting the 2001:7::/64, 2001:2::/64 routes to pass.

```
Device3(config)#ipv6 prefix-list 7001 permit 2001:7::/64
Device3(config)#ipv6 prefix-list 7002 permit 2001:2::/64
```

Configure the routing policy lp so that Device3 allows setting local-preference for routes that match the prefix list 7001.

```
Device3(config)#route-map lp 10
Device3(config-route-map)#match ipv6 address 7001
Device3(config-route-map)#set local-preference 200
Device3(config-route-map)#exit
Device3(config)#route-map lp 20
Device3(config-route-map)#exit
```

Configure the routing policy med so that Device3 allows setting the MED attribute for routes that match the prefix list 7002.

```
Device3(config)#route-map med 10
Device3(config-route-map)#match ipv6 address 7002
Device3(config-route-map)#set metric 10
Device3(config-route-map)#exit
Device3(config)#route-map med 20
Device3(config-route-map)#exit
```

Note:

- When configuring the routing policy, both the prefix list and ACL can create the match rule, and the difference is that the prefix list can correctly match the route mask, but the ACL cannot match the route mask.

Step 5: Configure BGP to associate the routing policy.

#Configure Device2.

Apply the routing policy lp to the outgoing route of the neighbor 1:1, and apply the routing policy med to the outgoing route of the neighbor 2001 :5 ::2.

```
Device2(config)#router bgp 100
Device2(config-bgp)#address-family ipv6 unicast
Device2(config-bgp-af)#neighbor 1::1 route-map lp out
Device2(config-bgp-af)#neighbor 2001:5::2 route-map med out
```

```
Device2(config-bgp-af)#exit-address-family
Device2(config-bgp)#exit
```

#Configure Device3.

Apply the routing policy lp to the outgoing route of the neighbor 1:1, and apply the routing policy med to the outgoing route of the neighbor 2001 :6 ::2.

```
Device3(config)#router bgp 100
Device3(config-bgp)#address-family ipv6 unicast
Device3(config-bgp-af)#neighbor 1::1 route-map lp out
Device3(config-bgp-af)#neighbor 2001:6::2 route-map med out
Device3(config-bgp-af)#exit-address-family
Device3(config-bgp)#exit
```

Step 6: Check the result.

#View the BGP route information of Device1.

```
Device1#show bgp ipv6 unicast
BGP table version is 5, local router ID is 1.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
          S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
    Network          Next Hop         Metric    LocPrf Weight Path
[B]*> 2001:1::/64       ::                0          32768 i
[B]*> 2001:2::/64       ::                0          32768 i
[B]* i2001:7::/64       2::2              0       100      0 200 i
[B]*>i                  3::3              0       200      0 200 i
[B]*>i2001:8::/64       2::2              0       200      0 200 i
[B]* i                  3::3              0       100      0 200 i

Total number of prefixes 6
```

#View the route table of Device1.

```
Device1#show ipv6 route bgp
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
     U - Per-user Static route
     O - OSPF, OE-OSPF External, M - Management

B   2001:7::/64 [200/0]
    via 3::3, 00:11:49, gigabitethernet1
B   2001:8::/64 [200/0]
```

```
          via 2::2, 00:05:24, gigabitethernet0
```

According to the IPv6 BGP route table of Device1, the route 2001:7::/64 has two next hops, that is, 2::2 and 3::3, while the local priority of the route with the next hop 3:3 changes to 200, so the data to the network segment 2001:7::/64 first chooses to pass Device3; the route 2001:8::/64 also has two next hops, that is, 2::2 and 3::3, while the local priority of the route with the next hop 2::2 changes to 200, so the data to the network segment 2001:8::/64 first chooses to pass Device2.

#View the BGP route information of Device4.

```
Device4#show bgp ipv6 unicast
BGP table version is 6, local router ID is 4.4.4.4
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
          S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
    Network           Next Hop          Metric    LocPrf Weight Path
[B]*  2001:1::/64       2001:5::1           10            0 100 i
[B]*>                2001:6::1          0             0 100 i
[B]*> 2001:2::/64       2001:5::1           0             0 100 i
[B]*                 2001:6::1          10            0 100 i
[B]*> 2001:7::/64       ::                 0          32768 i
[B]*> 2001:8::/64       ::                 0          32768 i

Total number of prefixes 6
```

#View the route table of Device4.

```
Device4#show ipv6 route bgp
Codes: C - Connected, L - Local, S - static, R - RIP,  B - BGP, i-ISIS
     U - Per-user Static route
     O - OSPF, OE-OSPF External, M - Management

B   2001:1::/64 [20/0]
    via 2001:6::1, 00:18:57, gigabitethernet1
B   2001:2::/64 [20/0]
    via 2001:5::1, 00:10:11, gigabitethernet0
```

According to the IPv6 BGP route table of Device4, the route 2001:1::/64 has two next hops, that is, 2001:5::1 and 2001:6::1, while the metric of the route with the next hop 2001:6::1 changes to 10, so the data to the network segment 2001:1::/64 first chooses to pass Device3; the route 2001:2::/64 also has two next hops, that is, 2001:5::1 and 2001:6::1, while the metric of the route with the next hop 2001:5::1 changes to 10, so the data to the network segment 2001:2::/64 first chooses to pass Device2.

Note:

- When the BGP peer or peer group applies the routing policy, it can be used at the receiving and advertising direction of the peer or peer group. It takes effect after resetting BGP.

16. ОБЩАЯ ИНФОРМАЦИЯ

16.1. Замечания и предложения

Мы всегда стремимся улучшить нашу документацию и помочь вам работать лучше, поэтому мы хотим услышать вас. Мы всегда рады обратной связи, в особенности:

- ошибки в содержании, непонятные или противоречащие места в тексте;
- идеи по улучшению документации, чтобы находить информацию быстрее;
- неработающие ссылки и замечания к навигации по документу.

Если вы хотите написать нам по поводу данного документа, то используйте, пожалуйста, форму обратной связи на qtech.ru.

16.2. Гарантия и сервис

Процедура и необходимые действия по вопросам гарантии описаны на сайте QTECH в разделе «Поддержка» −> «Гарантийное обслуживание».

Ознакомиться с информацией по вопросам тестирования оборудования можно на сайте QTECH в разделе «Поддержка» −> «Взять оборудование на тест».

Вы можете написать напрямую в службу сервиса по электронной почте sc@qtech.ru.

16.3. Техническая поддержка

Если вам необходимо содействие в вопросах, касающихся нашего оборудования, то можете воспользоваться нашей автоматизированной системой запросов технического сервис-центра helpdesk.qtech.ru.

Телефон Технической поддержки +7 (495) 797-33-11 доб. 0