Contents

1 Configuring BGP

1.1 Introduction

1.1.1 Overview

The Border Gateway Protocol (BGP) is a dynamic routing protocol that automatically exchanges loop-free routing information between autonomous systems (ASs).

A network is divided into multiple ASs to facilitate management. In the early Internet, the Exterior Gateway Protocol (EGP) was used to switch routing information between ASs. This protocol was first mentioned in RFC 187 and then listed into RFC 904 in 1984. EGP was a simple network reachability protocol and applied only to tree networks. Therefore, it could not meet the increasingly complicated network management requirements and finally was replaced by BGP. Unlike EGP, BGP can avoid loops and select better routes, and efficiently transfers routes and supports higher scalability.

The development of BGP has undergone different stages, and BGP-1 (RFC 1105), BGP-2 (RFC 1163), and BGP-3 (RFC 1267) were previously released. The prevailing version is BGP-4 (RFC 4271), which is widely applied between ASs as a standard routing protocol.

BGP has the following features:

- Interior Gateway Protocols (IGPs) such as Open Shortest Path First (OSPF) and Routing Information Protocol (RIP) focus on discovery and route calculation. By contrast, BGP lays emphasis on the control of route spreading and is applicable to networks that carry massive routes.
- BGP uses Transmission Control Protocol (TCP) as the transmission protocol and the destination port is 179. BGP uses the reliable transmission mechanism of TCP to ensure transmission reliability.
- As a distance vector routing protocol, BGP transmits only the locally selected optimal routes to its neighbors.

Note

In this chapter, the term "router" refers to any network device that supports the routing function. These network devices can be Layer-3 (L3) switches, routers, or firewalls.

1.1.2 Concepts

1. BGP speaker

Devices that run BGP and send BGP packets are called BGP speakers.

2. BGP peer

BGP speakers with a BGP session established are BGP peers. Multiple peers constitute a peer group.

3. AS

Devices that manage and share the same routing policies in one organization are an AS. When BGP is enabled on a router, a local AS number must be specified for it. An AS number is a globally unique number allocated by IANA, consisting of two or four bytes.

A traditional AS number consists of 2 bytes, ranging from 1 to 65535. The new 4-byte AS numbers aims to resolve the resource exhaust problem, and their value range is from 1 to 4294967295.

4-byte AS numbers support two expression modes: decimal notation mode and dot mode. The decimal notation mode is the same as the traditional expression mode, that is, the 4 bytes of an AS number are expressed in decimal notation. The conversion relationship between the dot mode and decimal notation mode is as follows: 4-byte decimal AS number = x × 65536 + Y.

For example:

- For an AS number 65534 in decimal mode, it is 65,534 in dot mode. An AS number smaller than 65536 is the same in decimal mode and dot mode.
- For an AS number 65536 in decimal mode, it is 1.0 in dot mode.
- For an AS number 65538 in decimal mode, it is 1.2 in dot mode.

With introduction of 4-byte AS numbers, BGP connections may be established between BGP speakers supporting only 2-byte AS numbers and those supporting 4-byte AS numbers. To communicate with a device that supports 4-byte AS numbers, reserve the AS number 23456 for filling on the device that supports only 2-byte AS numbers.

4. Neighbor type and route type

BGP neighbors are classified into two types:

- Internal BGP (IBGP) neighborship: The neighborship between BGP speakers within an AS is called IBGP neighborship. Routes learned from IBGP neighbors are called IBGP routes. IBGP completes transition of routing information within the AS.
- External BGP (EBGP) neighborship: The neighborship between BGP speakers in different ASs is called EBGP neighborship. Routes learned from EBGP neighbors are called EBGP routes. EBGP is used to exchange routing information between different ASs.

1.1.3 Route Attributes

When a BGP speaker advertises a route to its neighbors, the BGP speaker also advertises the attributes carried by the route. Common BGP attributes are as follows:

1. Origin attribute

Specifies the origin of a BGP route, which is set to IGP, EGP, or INCOMPLETE.

- IGP: The value is 0, specifying the internal routing information of the originating AS.
- EGP: The value is 1, specifying the routing information obtained from EGP.
- INCOMPLETE: The value is 2, specifying the routing information learned through other means.

2. AS_PATH attribute

Lists the ASs passed by a route reversely. The last AS is placed at the beginning of the list.

The AS_PATH attribute is used for:

- When a router receives routing information with the local AS number in the AS_PATH, the router considers that this route has loop and discards the packet.
- Routing information distribution is controlled by using the AS_PATH-based filtering rules. These filtering rules use regular expressions to parse AS paths.
- According to RFC 1771, BGP does not consider the AS path length when selecting an optimal path. Generally, a shorter AS path means a higher path priority; therefore, devices consider the AS path length when selecting the optimal path. Based on the actual condition, you can determine whether to consider the AS path length when selecting the optimal path.

3. NEXT_HOP attribute

Specifies the IP address of the next hop to be reached by a BGP route.

The change in the NEXT_HOP attribute complies with the following rules:

- When the route forwarded to an IBGP peer is non-local, the device does not change the NEXT_HOP attribute.
- When the route forwarded to an IBGP peer is local, the local interface address that the device uses to establish a BGP neighbor relationship with the peer is the next hop address.
- When the route is forwarded to an EBGP peer, the local interface address that the device uses to establish a BGP neighbor relationship with the peer is the next hop address.

4. MULTI_EXIT_DISC attribute

Distinguishes multiple output or input points for reaching the same neighbor AS. A smaller value means a higher priority. The MULTI_EXIT_DISC attribute is short for MED.

BGP uses the MED value as the basis for comparing priorities of paths learned from EBGP peers. A smaller MED value means a higher path priority.

- The **MED** value is compared only for paths of peers from the same AS during the optimal path selection by default.
- The **MED** value is not compared for paths of peers from other member ASs in an AS alliance during the optimal path selection by default.
- If a path not configured with the MED attribute is received, the MED value of this path is **0** by default. Since a smaller MED value means a higher path priority, the MED value of this path has the highest priority.
- The **MED** value is not compared for paths from different ASs during the optimal path selection by default. Instead, the receiving sequence of the paths is compared.

5. LOCAL_PREF attribute

Distinguishes the priorities of IBGP routes in an AS. A larger value means a higher priority.

When sending routes received from EBGP peers to IBGP peers, a BGP speaker adds the LOCAL_PREF attribute. BGP uses the LOCAL_PREF value as the basis for comparing priorities of paths learned from IBGP peers. A larger LOCAL_PREF value means a higher path priority.

6. COMMUNITY attribute

The **COMMUNITY** attribute is another method of controlling the distribution of routing information.

A community is a set of destination addresses. The **COMMUNITY** attribute is intended to facilitate execution of a community-based routing policy so as to simplify the configuration of routing information distribution control on BGP speakers. By using the community attribute, you can control the receiving and distribution of routing information. A destination address can belong to multiple communities. An AS administrator can define the communities, to which a destination address belongs.

BGP pre-defines four common community attribute values:

- **Internet**: Indicates the Internet community. All paths belong to this community.
- **No-export**: Specifies that the path is not advertised to EBGP peers.
- **No-advertise**: Specifies that the path is not advertised to any BGP peer.
- **Local-as**: Indicates that a path is not advertised outside the local AS. When an alliance is configured, the path is not advertised to other ASs or member ASs.

By default, all destination addresses belong to the Internet community and are carried in the **COMMUNITY** attribute of paths.

BGP speakers can set, add, or modify the community attribute value when learning, advertising, or re-distributing routes. An aggregate path will contain the community attribute values of all aggregated paths.

7. AIGP metric attribute

The AIGP metric attribute can accumulate the IGP metrics of paths that a route passes through. Preferred route selection based on the AIGP attribute value can reflect the path cost more accurately. A smaller AIGP attribute value indicates a higher route priority. Routes carrying the AIGP attribute have a higher priority than those without the AIGP attribute.

If the next hop of a route carrying the AIGP attribute is changed to self or another value during route transmission, the IGP metric value of the route to the original next-hop address needs to be added to the AIGP attribute value.

1.1.4 Route Exchange and Route Selection Policy of BGP

1. Route exchange

BGP cannot automatically discover or learn accessible networks. The accessible network information of a local AS must be imported to BGP so that BGP can advertise this information to neighbors.

Two methods can be used to import network information of the local AS to BGP:

- Manual configuration: Import accessible network information within a specified range to BGP.
- Configuring route re-distribution: Configure route re-distribution to re-distribute accessible network information of IGP to BGP.

BGP offers a robust route management function. By configuring a route exchange policy for a BGP peer, you can actively control routes to be received by and to be advertised to this peer.

Send routing information of other ASs exchanged by BGP to the routing table of a device so that the device can forward packets of other ASs. Import routing information of other ASs exchanged by BGP to IGP so that IGP distributes the information to other IGP devices in the local AS and these devices can forward packets of other ASs.

Generally, BGP speakers serving as IBGP neighbors of each other are not directly connected physically. IGP devices in the link between the BGP speakers may fail to learn routing information same as that on the BGP

speakers. When a BGP speaker at the border of an AS forwards packets received from other domains to the next-hop IBGP neighbor, the packets pass an IGP device in the middle. In this case, the packets may be lost due to no routing information on the IGP device. To synchronize BGP with IGP, you must ensure that all routers within an AS learn routing information to be sent to another AS before the routing information is advertised to this AS. BGP and IGP should be synchronized in most cases, except that:

- Routing information passing through this AS is not configured. For example, the AS is a stub AS.
- All routers within the AS run BGP. Full mesh is established among all BGP speakers (neighborship is established between each two BGP speakers).

2. BGP route update group

The BGP route update group function is a technology used to enhance the performance for advertising routes to neighbors.

The BGP route update group technology automatically classifies neighbors using the same policy into the same update group. When sending a route to a neighbor, the device encapsulates the update packet based on the update group, and sends the update packet to all neighbors in the update group. In this way, the update packet is encapsulated only once but sent multiple times, improving the performance of route advertisement to neighbors.

3. Route selection policy

BGP policies for selecting the optimal route are:

(1) Invalid routing entries (entries with the next hop unreachable and flapping entries) are not used for optimal route selection.

(2) If not, the route generated by the **network** command is preferred.

(3) Otherwise, a route with the maximum weight is preferred (the default weight value of a route originated locally is **32768**, and the weight value of a router advertised by a neighbor is **0**).

(4) If routes have the same weight, a route with a larger **LOCAL_PREF** value is preferred.

(5) If routes have the same local priority, a local BGP route from the local router is preferred.

(6) The types of local routes are compared, and the route priority is as follows: direct route > static route > aggregate route generated by BGP.

(7) If the local routes have the same type, the Accumulated Interior Gateway Protocol (AIGP) extended route is compared. A route that carries the AIGP attribute is prior to that without the AIGP attribute.

(8) If the AIGP attribute comparison is skipped, the link state (LS) extended attribute is compared:

a A route with the LS attribute is preferred.

b A route with a smaller IGP metric is preferred.

c A route of the wide type has a higher priority than that of the narrow type.

(9) Otherwise, a route with the shortest AS length is selected.

(10) Otherwise, a route with the lowest **ORIGIN Code** attribute value is selected.

(11) Otherwise, a route with the smallest **MED** value is selected.

(12) Otherwise, EBGP routes have a higher priority than IBGP routes and routes in an AS alliance, and the IBGP routes have the same priority as the routes in an AS alliance.

(13) Otherwise, a route with the smallest IGP metric value to the next hop is selected.

(14) Otherwise, an EBGP route that is received first is selected.

(15) Otherwise, a route advertised by a BGP speaker with a smaller router-ID is selected.

(16) Otherwise, a route with a smaller cluster length is selected.

(17) Otherwise, a route with a smaller neighbor address is selected.

1.1.5 Neighbor Connection Establishment

A BGP neighbor is manually configured by a user. Neighbor relationships can be established in two modes: IBGP and EBGP. Users can identify the connection mode between BGP speakers according to the AS carrying the BGP peer and the AS carrying the BGP speaker.

A BGP speaker proactively sends a TCP connection request to a specified BGP peer. After a TCP connection is established successfully, BGP speakers exchange BGP protocol packets to negotiate about connection parameters. The BGP neighbor relationship is successfully established after the negotiation succeeds.

1. Establishing a TCP connection

A BGP speaker initiates a TCP connection request to a neighbor. The destination IP address of the request is the peer IP address specified by the user and the port number is fixed to **179**.

The BGP speaker also listens on port 179 of the local TCP connection to receive connection requests from its peers.

2. Negotiating about protocol parameters

After a TCP connection is successfully established, BGP speakers exchange OPEN packets to negotiate about BGP connection parameters. Negotiated parameters include the following:

- **Version**: Indicates the BGP version number. At present, only version 4 is supported.
- **Neighbor AS number**: Determines whether the AS number of a neighbor is consistent with the AS number specified locally. If no, the connection request is denied.
- **Hold time**: Indicates the timeout duration for a BGP connection. The default value is **180** seconds.
- **Neighbor capability**: Negotiates about various extended capabilities supported by the neighbor, including the address family, dynamic route update and GR functions.

3. Maintaining the neighbor relationship

BGP speakers send the Keepalive message to each other periodically. If a new Keepalive packet is not received from a BGP neighbor after the hold time expires, the BGP speaker considers the neighbor unreachable, disconnects the TCP connection from the neighbor, and attempts to reconnect to it. The interval for a BGP speaker to send the Keepalive message is one third of the negotiated hold time and is **60** seconds by default.

The software allows you to manually configure various timers within BGP to meet the neighbor keepalive and route management requirements in different network environments.

- BGP neighbor keepalive timer

  BGP uses the keepalive timer to maintain a valid connection with a peer and the holdtime timer to judge whether the peer is valid. The default value of the keepalive timer is **60** seconds and the default value of the holdtime timer is **180** seconds. When BGP speakers establish a BGP connection, both parties

negotiate about the holdtime and the smaller holdtime value is selected. The keepalive timer is set to one third of the negotiated holdtime value or the configured keepalive value, whichever is smaller.

- Neighbor reconnection timer

  To reduce the impact of frequent BGP reconnection to a failed neighbor on the network bandwidth, after a BGP speaker detects a neighbor connection failure, it attempts to reconnect to the neighbor after the Connect-Retry timer expires. The default value of the Connect-Retry timer is **15** seconds.

- Route advertisement timer

  To reduce the impact of route update packets on the network bandwidth, after a BGP speaker detects a network topology change, it does not advertise the route update to its neighbors immediately. Instead, the BGP speaker adopts a regular update mechanism to advertise all changed routing information to its neighbors.

1.1.6 Multi-path Load Balancing of BGP

Multi-path load balancing means that there are multiple paths to the same network and data packets are evenly forwarded by these paths. In this case, one route has multiple next hops in the routing table.

According to the types of equal-cost routes, multi-path load balancing of BGP is classified into the following types:

- EBGP load balancing: Implements load balancing for routes learned from EBGP neighbors.
- IBGP load balancing: Implements load balancing for routes learned from IBGP neighbors.

When there are multiple paths to the same network in the routing table of BGP, BGP calculates a route with the highest priority by default. If there are multiple optimal routes with the same priority, BGP still selects a unique route according to comparison rules, advertises the route to the forwarding plane, and controls the forwarding of data streams. After multi-path load balancing is enabled, in addition to calculating a unique optimal route, BGP lists paths with the same priority as the optimal route as equal-cost routes. Then, BGP advertises the optimal route and equal-cost routes to the forwarding plane to implement load balancing.

Equal-cost routes have the same basic attributes and priority. That is, according to the optimal path selection rules of BGP, paths have the same priorities before router-IDs are compared.

- AS_PATH loose comparison

  Equal-cost routes must have the same AS-PATH attribute by default. Under such strict conditions, load balancing cannot be implemented in certain environments. In this case, you are advised to enable the AS-PATH loose comparison mode. In AS-PATH loose comparison mode, when other conditions for equal-cost routes are met, equal-cost route conditions are met as long as the AS-PATH lengths of routes and the AS-PATH lengths of the member ASs are equal respectively.

- Router-ID multi-path comparison

  Equal-cost routes do not need to come from the same device (that is, the sources of the routes do not need to have the same router ID) by default. After the router-ID multi-path comparison is enabled, only routes with the same router-ID can be equal-cost routes.

- Non-equal-cost load balancing

  Load-balanced routes are equal-cost routes by default. After the non-equal-cost load balancing is enabled, you can configure routes to carry the Link-Bandwidth attribute so that non-equal-cost routes form.

1.1.7 Route Reflector

Route reflectors are a method of reducing IBGP peer connections in an AS.

According to the principle of BGP route advertisement, all BGP speakers in an AS must establish a full mesh of connections (every two BGP speakers need to establish a neighbor relationship). Too many BGP speakers in an AS will increase their resource cost, ramp up network administrators' configuration workload and complexity, and reduce the network expansion capacity.

You can configure a BGP speaker as a route reflector, which classifies IBGP peers in the AS into two types: clients and non-clients.

The rules for implementing a route reflector in an AS are as follows:

- Configure a route reflector and specify clients for the route reflector. The route reflector and its clients form a cluster. The route reflector will connect to its clients.
- The clients of a route reflector in a cluster cannot connect to BGP speakers outside the cluster.
- Within an AS, a full mesh of connections is established among IBGP peers of non-clients. The IBGP peers of non-clients include the following situations: Multiple route reflectors in a cluster; a route reflector in a cluster and BGP speakers (generally not supporting the route reflector function) not involved in the route reflector function out of the cluster; a route reflector in a cluster and route reflectors in other clusters.

The rules for processing a route received by a route reflector are as follows:

- A route update message received from an EBGP speaker will be sent to all clients and non-clients.
- A route update message received from a client will be sent to other clients and all non-clients.
- A route update message received from an IBGP non-client will be sent to all clients of the route reflector.

1.1.8 BGP Alliance

Alliances are another method of reducing IBGP peer connections in an AS.

You can divide an AS into multiple member ASs and configure a unified alliance ID (namely, alliance AS number) for these member ASs to form an alliance. Outside the alliance, the entire alliance is still considered as an AS and only the AS number of the alliance is visible. Inside the alliance, BGP speakers in a member AS still establish a full mesh of IBGP peer connections, BGP speakers in different member ASs establish EBGP connections. Though EBGP connections are established between BGP speakers in different member ASs, the NEXT_HOP, MED, LOCAL_PREF, and other path attributes keep unchanged during information exchange.

1.1.9 Route Aggregation

The great BGP routing table in a large network brings heavy load to BGP devices. BGP supports classless inter-domain routing (CIDR). Therefore, route aggregation entries can be created to reduce the size of the BGP routing table. BGP aggregation entries can be added to the BGP routing table only when valid paths are configured within the aggregation range. Routes can be manually aggregated on a device. Thus, this function can control whether to reserve AS path information within the aggregation address range, and whether to advertise only aggregated paths.

1.1.10 Route Dampening

If a route changes between the valid state and invalid state, route flapping occurs. Route flapping often causes the transmission of unstable routes in a network, and thereby causes network instability. BGP route dampening

is a method of reducing route flapping. It reduces possible route flapping by monitoring routing information from EBGP peers.

Terms involved in BGP route dampening are as follows:

- **Route Flap**: A route changes between the valid sate and invalid state.
- **Penalty**: Once route flapping occurs, a BGP speaker, on which route dampening is enabled, adds a penalty value to this route. The penalty value accumulates until **Suppress Limit** is reached.
- **Suppress Limit**: When the penalty of a route is greater than this value, the route is suppressed.
- **Half-life-time**: It indicates the time required for the penalty to be halved.
- **Reuse Limit**: When the penalty value of a route is smaller than this value, route suppression is canceled.
- **Max-suppress-time**: It indicates the maximum time that a route can be suppressed.

A BGP speaker assigns a route a penalty (accumulated to the penalty value) each time the route flaps. When the penalty value reaches **Suppress Limit**, the route is suppressed. When the penalty value reaches **Half-life-time**, the penalty is halved. When the penalty value is reduced to **Reuse Limit**, the route is activated again. The maximum time that a route can be suppressed is the value of **Max-suppress-time**.

By default, the penalty value increases by 1000 when a route is reactivated and increases by 500 when a route is updated.

1.1.11 BGP Reliability

1. GR

Graceful restart (GR) is intended to ensure that data forwarding is not interrupted during the BGP restart. The GR function keeps the network topology stable, maintains the forwarding table, and ensures that key services are not interrupted during the switching of master and slave supervisor engines.

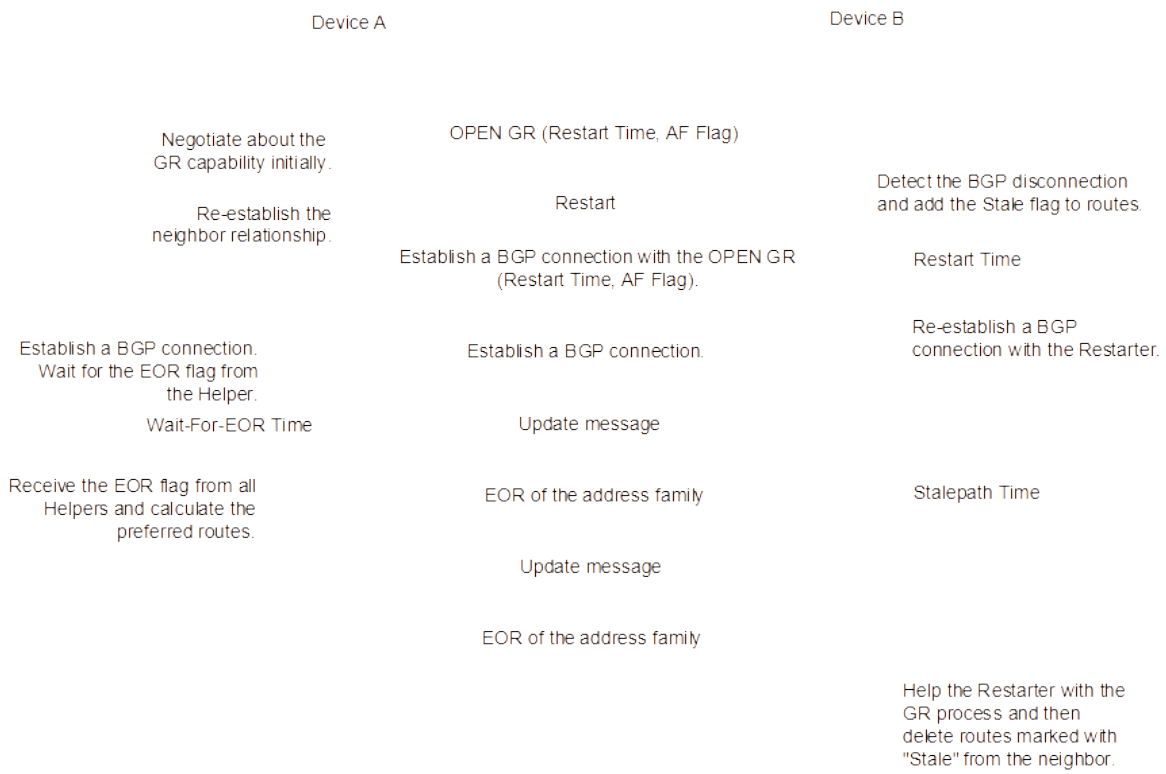
BGP GR is not an independent process, but is jointly accomplished by the Restarter and Helper.

- The Restarter performs restart and is capable of making the route forwarding plane to keep working when the route control plane is faulty.
- The Helper is the BGP neighbor of the Restarter and helps the Restarter complete GR.

To deliver graceful restart, BGP adds a capability field indicating to neighbors its support for GR to the OPEN message. The capability is called Graceful Restart Capability. During initialization of a BGP connection, both neighbors negotiate about the GR capability.

The End-of-RIB (EOR) flag is added to the update packets of BGP, to indicate that the routing information update to the neighbor is completed.

Figure 1-1 BGP GR Interaction Process



(1) When BGP establishes a neighbor relationship at the beginning, BGP uses the GR capability field in the OPEN message to negotiate about the GR capabilities of both neighbors.

(2) At a moment, the Restarter performs restart, and the BGP session is disconnected. The Helper detects the disconnection, keeps the route of the Restarter valid, but marks the Stale (aged but not updated).

(3) The Restarter completes restart.

(4) The Restarter connects to the Helper again.

(5) The Restarter waits for the route update message and EOR flag from the Helper.

(6) After receiving the EOR flag from all neighbors, the Restarter performs route calculation, updates routing entries, and then sends updated routes to the Helper.

(7) After receiving the updated routes, the Helper cancels their Stale flag. After receiving the EOR flag from the Restarter, the Helper deletes routes with the Stale flag (these routes are not updated), calculates routes calculation, and updates the routing entries. The entire GR process is completed.

BGP GR defines several important extended timers:

- **Restart-Timer**

  The value of this timer is advertised by the GR Restarter to the GR Helper. It indicates the maximum waiting time that the GR Restarter hopes the Helper to wait before a new connection is established between them. You can run the **bgp graceful-restart restart-time** command to change the time value.

- **Wait-For-EOR Timer**

  This timer indicates the maximum time for the GR Restarter to wait for the EOR flag from all GR Helpers. After the EOR flag is received from all GR Helpers or the Wait-For-EOR timer expires, the GR Restarter calculates the preferred routes and updates the routing entries.

- **StalePath Timer**

  The timer indicates the maximum time for the GR-Helper to wait for the EOR flag from the GR Restarter after the connection between them is restored. Within this period, the Helper keeps the original route of the Restarter valid. After receiving the EOR flag or after the StalePath timer expires, the Helper clears the routing entries with the Stale tag. You can run the **bgp graceful-restart stalepath-time** command to change the time value.

2. BGP NSR

None-stop-routing (NSR) is used to ensure uninterrupted routes during protocol restart upon a switchover between the master and slave supervisor engines. During the switchover, the NSR function keeps the network topology stable, maintains the neighbor status and forwarding table, and ensures that key services are not interrupted.

Unlike BGP GR, BGP NSR does not need the assistance of neighbors.

- Connections between BGP neighbors are not interrupted during active/standby switchover.
- Neighbors do not perceive the restart behavior of the peer device. The GR technology is used for restart restoration.

Neighbors with the BGP NSR function enabled do not need to negotiate about the GR capability nor need assistance from the peer. Therefore, no interworking problem arises.

3. Fast rerouting

With the fast development of IP technologies and the application of various complex services, higher requirements are posed for network security and stability. Some real-time services (audios and videos) are sensitive to the network running status and are largely affected by unstable networks. Therefore, network reliability receives more and more attention. In response to this requirement, the IP Fast Reroute (FRR) function appears. This function aims to use a backup link to forward data during route platform convergence after a faulty link is detected so as to ensure that intact packets are forwarded timely.

Thus, BGP FRR can avoid route disconnection due to a link fault before BGP route convergence. If a BGP routing table has multiple paths to the same network, BGP calculates the route with the highest priority by default. After the BGP FRR function is used, BGP selects a backup route for each optimal route. When detecting a fault occurring on the master link by using the BFD fast link detection mechanism, the device switches data to the selected backup link for data forwarding. After route convergence is completed, the device switches data to the re-calculated optimal path for data forwarding.

1.1.12 BMP

BGP Monitor Protocol (BMP) monitors BGP running on a device. It focuses on the status of BGP peers, peer entry transmission and receiving, and all kinds of BGP peer statistics.

A BGP speaker proactively sends a TCP connection request to a specified BMP server. After a TCP connection is established successfully, the BGP speaker advertises a BMP packet to the BMP server. In the TCP connection request initiated by a BGP speaker to a BMP server, the destination IP address is the IP address of the specified BMP server and the port number is a specified port number.

1.1.13 Others

1. Regular expression

A regular expression matches strings based on a rule. It is used to assess text data and return True or False to decide whether the expression can correctly describe the data.

The following table describes special characters involved in regular expressions used in BGP path attributes.

Table 1-1 Description of Special Characters in a Regular Expression

Character	Symbol	Special Meaning
Period	.	Matches any single character.
Asterisk	*	Matches zero or any sequence in a string.
Plus sign	+	Matches one or any sequence in a string.
Question mark	?	Matches zero or one symbol in a string.
Caret	^	Matches the start of a string.
Dollar sign	$	Matches the end of a string.
Underline	_	Matches commas, brackets, start and end of a string, and spaces.
Square brackets	[ ]	Matches a single character within a range.

1.1.14 Protocols and Standards

- RFC 1105: Border Gateway Protocol (BGP)
- RFC 1163: Border Gateway Protocol (BGP)
- RFC 1267: Border Gateway Protocol 3 (BGP-3)
- RFC 1771: A Border Gateway Protocol 4 (BGP-4)
- RFC 4271: A Border Gateway Protocol 4 (BGP-4)
- RFC 4273: Definitions of Managed Objects for BGP-4
- RFC 4360: PrBGP Extended Communities Attribute
- RFC 4364: Proposed Standard: BGP/MPLS IP Virtual Private Networks (VPNs)
- RFC 4486: Proposed Standard: Subcodes for BGP Cease Notification Message
- RFC 4724: Proposed Standard: Graceful Restart Mechanism for BGP
- RFC 4760: Draft Standard: Multiprotocol Extensions for BGP-4
- RFC 4893: BGP Support for Four-octet AS Number Space
- RFC 5396: Textual Representation of Autonomous System (AS) Numbers
- RFC 5492: Draft Standard: Capabilities Advertisement with BGP-4
- RFC 6198: Requirements for the Graceful Shutdown of BGP Sessions

- RFC 7313: Enhanced Route Refresh Capability for BG: P-4
- RFC 7311: The Accumulated IGP Metric Attribute for BGP
- RFC 7432: Proposed Standard: BGP MPLS-based Ethernet VPN

1.2 Configuration Task Summary

BGP configuration includes the following tasks:

- Configuring BGP Basic Features
  - Configuring a BGP Neighbor
  - (Optional) Configuring a BGP IPv6 Address Family
  - (Optional) Configuring a BGP Peer Group
  - (Optional) Configuring the Dot Mode to Display 4-Byte AS Numbers
  - Configuring Routing Information Advertisement
  - (Optional) Configuring Multi-Path Route Import Between VRF Instances
  - (Optional) Shutting Down BGP Connections Gracefully
  - (Optional) Configuring BGP Soft Reset
  - (Optional) Configuring Route Update Mechanisms of BGP
  - (Optional) Configuring BGP Capacity Protection
- (Optional) Configuring a large BGP Network. All the following configuration tasks are optional and may be selected as needed.
  - Configuring a Route Reflector
  - Configuring an AS Alliance
  - Configuring Route Aggregation
  - Configuring Route Dampening
- (Optional) Configuring BGP Route Selection and Load Balancing. All the following configuration tasks are optional and may be selected as needed.
  - Configuring Multi-path Load Balancing of BGP
  - Configuring BGP Routes to Be Recursive Only to Host Routes
  - Configuring Outbound Loop Detection for a BGP Neighbor
  - Configuring BGP ADD-PATH
  - Configuring BGP to Advertise Routes with the Lowest Priority upon Device Restart
  - Configuring the AS_PATH Attribute
- (Optional) Configuring BGP to Control Route Advertisement and Receiving. All the following configuration tasks are optional and may be selected as needed.
  - Configuring Fast Withdrawal of Specified BGP Routes
  - Configuring Delayed Route Advertisement of BGP
- (Optional) Configuring the BGP Security Function. All the following configuration tasks are optional and may be selected as needed.

- Configuring MD5 Authentication
- Configuring the Generalized TTL Security Mechanism (GTSM) Security Check for BGP Neighbors

- (Optional) Configuring BGP Reliability. All the following configuration tasks are optional and may be selected as needed.
  - Configuring BFD For BGP
  - Configuring BGP FRR
  - Configuring EBGP Multi-Path Bypass Protection
  - Configuring BGP GR
  - Configuring BGP NSR
  - Configuring BGP Session Retention
- (Optional) Enabling the Extended Functions of BGP. All the following configuration tasks are optional and may be selected as needed.
  - Configuring BMP Monitoring
  - Configuring the Administrative Distance of BGP

1.3 Configuring BGP Basic Features

1.3.1 Configuration Tasks

The configuration of BGP basic features includes the following tasks:

- Configuring a BGP Neighbor
- (Optional) Configuring a BGP IPv6 Address Family
- (Optional) Configuring a BGP Peer Group
- (Optional) Configuring the Dot Mode to Display 4-Byte AS Numbers
- Configuring Routing Information Advertisement
- (Optional) Configuring Multi-Path Route Import Between VRF Instances
- (Optional) Shutting Down BGP Connections Gracefully
- (Optional) Configuring BGP Soft Reset
- (Optional) Configuring Route Update Mechanisms of BGP
- (Optional) Configuring BGP Capacity Protection

1.3.2 Configuring a BGP Neighbor

1. Overview

This section describes how to configure a BGP speaker as a neighbor.

2. Restrictions and Guidelines

- If IBGP neighbors are not directly connected, you need to configure IGP or a static routing protocol to implement reachable routes between the neighbors.
- If EBGP neighbors are not directly connected, you need to configure the **ebgp-multihop** parameter for the neighbors.

- The source interface for establishing a neighbor relationship must be a valid local interface or IP address.
- You can run the **exit-address-family** command to exit the address family configuration mode of BGP.

3. Procedure

(1) Enter the privileged EXEC mode.

enable

(2) Enter the global configuration mode.

configure terminal

(3) Enable BGP to enter the BGP configuration mode.

router bgp *as-number*

(4) (Optional) Enter a required address family configuration mode. If you need to perform configuration for a specific address family, select one of the following tasks to configure.

- Enter the IPv4 unicast configuration mode.

  address-family ipv4 [**unicast**]

- Enter the IPv4 VRF configuration mode.

  address-family ipv4 vrf *vrf-name*

- Enter the IPv6 unicast configuration mode.

  address-family ipv6 [**unicast**]

- Enter the IPv6 VRF configuration mode.

  address-family ipv6 vrf *vrf-name*

(5) Create a BGP neighbor. When the configured AS number is the same as the local AS number, an IBGP neighbor is created. When the configured AS number is different from the local AS number, an EBGP neighbor is created.

neighbor { *neighbor-ipv4-address* | *neighbor-ipv6-address* | *peer-group-name* } **remote-as** *as-number*

No neighbor is specified for a BGP speaker by default.

(6) (Optional) Configure a source interface or IP address for establishing a TCP connection with the neighbor. A loopback interface is recommended for IBGP neighbors while a directly connected interface is recommended for EBGP neighbors.

neighbor { *neighbor-ipv4-address* | *neighbor-ipv6-address* | *peer-group-name* } **update-source** { *interface-type interface-number* | *address* }

BGP automatically selects the source IP address for a TCP connection based on the IP address of a neighbor by default. The IP address of the outbound interface of local packets is generally used.

(7) (Optional) Configure the time to live (TTL) for TCL packets received from the neighbor. A larger TTL value indicates more hops to the neighbor. When the TTL is **1**, the BGP neighbor must be directly connected to the device.

neighbor { *neighbor-ipv4-address* | *neighbor-ipv6-address* | *peer-group-name* } **ebgp-multihop** [*ttl*]

The default TTL values of TCP packets to be sent to IBGP neighbors and EBGP neighbors are **255** and **1** respectively.

(8) (Optional) Configure the mode for BGP to establish a TCP connection with the neighbor.

neighbor { *neighbor-ipv4-address* | *neighbor-ipv6-address* | *peer-group-name* } **transport connection-mode** { **active-only** | **both** | **passive-only** }

By default, BGP neighbors support the active and passive modes of establishing TCP connections.

(9) (Optional) Activate the address family capability for the neighbor.

neighbor { *neighbor-ipv4-address* | *neighbor-ipv6-address* | *peer-group-name* } **activate**

Only the IPv4 unicast address family capability is activated for neighbors created in BGP configuration mode by default.

1.3.3 Configuring a BGP IPv6 Address Family

1. Overview

This section describes how to configure BGP to exchange IPv6 routes to implement IPv6 network access between different ASs.

2. Restrictions and Guidelines

- Generally, BGP uses IPv6 addresses to establish neighbor relationships and implement the exchange of IPv6 routes.
- In special scenarios, for example, the IPv6 provider edge router (6PE) function is configured, BGP supports the exchange of IPv6 routes with the neighbors using IPv4 addresses.
- Configurations related to BGP IPv6 services must be completed in BGP IPv6 address family mode.
- Neighbors using IPv6 addresses are used to exchange IPv6 routes. When a neighbor is configured in BGP mode, BGP automatically activates the IPv4 unicast address family capability for the neighbor. You are advised to manually disable the IPv4 unicast address family capability.
- You can run the **exit-address-family** command to exit the address family configuration mode of BGP.

3. Procedure

(1) Enter the privileged EXEC mode.

enable

(2) Enter the global configuration mode.

configure terminal

(3) Enable BGP to enter the BGP configuration mode.

router bgp *as-number*

(4) Create a BGP neighbor.

neighbor { *neighbor-ipv4-address* | *neighbor-ipv6-address* | *peer-group-name* } **remote-as** *as-number*

(5) Configure the BGP IPv4 address family mode.

address-family ipv4 unicast

(6) Disable the IPv4 address family capability for the BGP neighbor.

no neighbor { *neighbor-ipv4-address* | *peer-group-name* } **activate**

The IPv4 address family capability is enabled by default.

(7) Configure the BGP IPv6 address family mode.

address-family ipv6 unicast

(8) Configure the IPv6 address family capability for the BGP neighbor.

neighbor { *neighbor-ipv6-address* | *peer-group-name* } **activate**

The IPv6 address family capability is disabled by default.

(9) (Optional) Configure BGP to advertise IPv6 routes.

network *network-ipv6-address/prefix-length* [**route-map** *route-map-name*] [**backdoor**]

No network routing information is configured for BGP by default.

1.3.4 Configuring a BGP Peer Group

1. Overview

BGP speakers in the same peer group have the same update strategy. Peer groups can simplify configuration and improve the running efficiency.

2. Restrictions and Guidelines

- Members of a peer group inherit all configurations of the peer group. Unified configuration of a peer group can be replaced by separate configuration of each member in the peer group, except the configuration that influences output update, for example, the configuration of neighbor distribute-list, neighbor route-map, and neighbor filter-list.
- You can neither put neighbors belonging to different address families to the same peer group nor put IBGP and EBGP neighbors to the same peer group.
- You can run the **exit-address-family** command to exit the address family configuration mode of BGP.

3. Procedure

(1) Enter the privileged EXEC mode.

enable

(2) Enter the global configuration mode.

configure terminal

(3) Enable BGP to enter the BGP configuration mode.

router bgp *as-number*

(4) (Optional) Enter a required address family configuration mode. If you need to perform configuration for a specific address family, select one of the following tasks to configure.

Enter the IPv4 unicast configuration mode.

address-family ipv4 [**unicast**]

- Enter the IPv4 VRF configuration mode.

  address-family ipv4 vrf *vrf-name*

- Enter the IPv6 unicast configuration mode.

  address-family ipv6 [**unicast**]

- Enter the IPv6 VRF configuration mode.

  address-family ipv6 vrf *vrf-name*

(5) Create a peer group.

neighbor *peer-group-name* **peer-group**

(6) Specify a BGP speaker as a member of the BGP peer group.

neighbor { neighbor-ipv4-address | neighbor-ipv6-address } **peer-group** *peer-group-name*

1.3.5 Configuring the Dot Mode to Display 4-Byte AS Numbers

1. Overview

The 4-byte AS numbers can be represented in dot mode.

2. Procedure

(1) Enter the privileged EXEC mode.

enable

(2) Enter the global configuration mode.

configure terminal

(3) Enable BGP to enter the BGP configuration mode.

router bgp *as-number*

(4) Configure the dot mode to display 4-byte BGP AS numbers.

bgp asnotation dot

1.3.6 Configuring Routing Information Advertisement

1. Overview

This section describes how to configure network information to be advertised by the local BGP speaker. Network information to be advertised can be direct routes, static routes, and dynamic routes.

2. Restrictions and Guidelines

- You are advised to enable the network information synchronization function. Otherwise, routing black holes may be incurred.
- You can run the **exit-address-family** command to exit the address family configuration mode of BGP.

3. Procedure

(1) Enter the privileged EXEC mode.

enable

(2) Enter the global configuration mode.

configure terminal

(3) Enable BGP to enter the BGP configuration mode.

router bgp *as-number*

(4) (Optional) Enter a required address family configuration mode. If you need to perform configuration for a specific address family, select one of the following tasks to configure.

- Enter the IPv4 unicast configuration mode.

  address-family ipv4 [**unicast**]

- Enter the IPv4 VRF configuration mode.

  address-family ipv4 vrf *vrf-name*

- Enter the IPv6 unicast configuration mode.

  address-family ipv6 [**unicast**]

- Enter the IPv6 VRF configuration mode.

  address-family ipv6 vrf *vrf-name*

(5) Add static routing entries to a BGP routing table and advertise the routing entries to peers.

network *network-number* [**mask** *mask*] [**route-map** *route-map-name*] [**backdoor**]

No static routing entry is added to a BGP routing table and advertised to peers by default.

(6) (Optional) Configure a device to advertise routing information configured by running the **network** command after the device synchronizes with the local route.

network synchronization

By default, a device will advertise routing information configured by running the **network** command after the device synchronizes with the local route.

1.3.7 Configuring Multi-Path Route Import Between VRF Instances

1. Overview

This section describes how to configure multi-path route import between VRF instances. Routes imported between VRF instances form multiple ECMP paths.

2. Restrictions and Guidelines

- This command does not allow IBGP and EBGP routes to form equivalent routes.

3. Procedure

(1) Enter the privileged EXEC mode.

enable

(2) Enter the global configuration mode.

configure terminal

(3) Enable BGP to enter the BGP configuration mode.

router bgp *as-number*

(4) Configure a BGP VRF address family.

(5) Enter a required address family configuration mode. If you need to perform configuration for a specific address family, select one of the following tasks to configure.

- Enter the IPv4 unicast configuration mode.

  address-family ipv4 [**unicast**]

- Enter the IPv4 VRF configuration mode.

  address-family ipv4 vrf *vrf-name*

- Enter the IPv6 unicast configuration mode.

  address-family ipv6 [**unicast**]

- Enter the IPv6 VRF configuration mode.

  address-family ipv6 vrf *vrf-name*

(6) Configure route re-distribution to re-distribute IGP routes to BGP.

redistribute *protocol-type* [**route-map** *route-map-name*] [**metric** *metric-value*]

The route re-distribution function is disabled by default.

(7) (Optional) Configure route re-distribution to re-distribute only IBGP routes to IGP.

bgp redistribute-internal

Routes learned from IBGP can be re-distributed to IGP by default.

(8) Enable the function of importing multiple next-hop routes from other protocols to BGP.

bgp sourced-paths *protocol-type* **all**

The function of importing multiple next-hop routes is disabled by default.

(9) Configure BGP ECMP multi-path load balancing.

maximum-paths { **ebgp** | **ibgp** } *maximum-path-number*

The ECMP function is disabled by default.

(10) Configure the import of routes of all paths between VRF instances, to control the import of all next-hop routes, preferred next-hop routes, or equal-cost next-hop routes.

import path selection { **all** | **bestpath** | **multipath** }

Only preferred next-hop routes are imported by default.

When routes are imported into a virtual routing and forwarding (VRF) instance, only routes with preferred next hops are imported. The enhanced VPN route import function extends the import of the preceding routes, and supports the import of all next-hop routes or equal-cost next-hop routes.

1.3.8 Shutting Down BGP Connections Gracefully

1. Overview

In a network, as a user isolates or upgrades a device, the RFC 6198 defines a function of shutting down BGP connections gracefully. This function demands the device to shut down connections with its neighbors when service traffic is not interrupted or is interrupted for a short time.

This feature adopts the "Make-Before-Break" mode to shut down BGP connections, to ensure that service traffic is not interrupted or is interrupted for a very short period of time in the process from the feature configuration to actual BGP connection shutdown. The steps are as follows:

(1) The feature enables the device to advertise a route with the lowest priority (the value of Local-Pref is **0** or the value of MED is 4,294,967,295), carrying Gshut Community, to a neighbor. Then, the neighbor updates the route and switches traffic to a backup link or other equal-cost links.

(2) The device shuts down the BGP connection with the neighbor after a period of time.

The delay time for BGP connection shutdown can be manually specified or automatically calculated. The automatically calculated delay time is based on 1 second for 1,000 routes.

2. Procedure

(1) Enter the privileged EXEC mode.

enable

(2) Enter the global configuration mode.

configure terminal

(3) Enable BGP to enter the BGP configuration mode.

router bgp *as-number*

(4) Shut down all connections in a BGP instance gracefully.

bgp shutdown graceful [**community** *community-value*] [**delay** *delay-time*]

None of the connections in a BGP instance is shut down by default.

(5) Create a BGP neighbor.

neighbor { *neighbor-ipv4-address* | *neighbor-ipv6-address* | *peer-group-name* } **remote-as** *as-number*

(6) Shut down connections of the BGP neighbor gracefully.

neighbor { *neighbor-ipv4-address* | *neighbor-ipv6-address* | *peer-group-name* } **shutdown graceful** [**community** *value*] [**delay** *time*]

None of the connections of a BGP neighbor is shut down by default.

1.3.9 Configuring BGP Soft Reset

1. Overview

If routing policies (including neighbor distribute-list, neighbor route-map, neighbor prefix-list, and neighbor filter-list) are modified, an effective method must be available to implement new routing policies. The conventional method is to configure BGP soft reset, which implements a new routing policy without terminating a BGP session connection. When an output routing policy is modified, BGP soft reset will re-advertise all routing information of a BGP speaker to its neighbors.

If an input routing policy is modified, the operation is more complex than that performed when an output routing policy is modified. This is because output routing policies are implemented on the routing table of the local BGP speaker, while the input routing policies are implemented on the routing information received from BGP peers. The local BGP speaker does not retain the original routing information received from BGP peers to reduce memory occupancy. If an input routing policy is modified and a neighbor supports the route update function, you can configure soft reset to send a route update request to the neighbor. After receiving the request, the neighbor re-advertises all routing information. You can also configure the local BGP speaker to store original routing information for each specified BGP peer and the original routing information is used as a basis for judging the modification of input routing policies.

2. Procedure

(1) Enter the privileged EXEC mode.

enable

(2) Clear the specified session of the BGP IPv4 unicast address family.

clear bgp ipv4 unicast [**vrf** vrf-name] { ***** | *as-number* | *neighbor-ipv4-address* | *neighbor-ipv6-address* } **soft**]**out**

(3) Enter the global configuration mode.

configure terminal

(4) Enable BGP to enter the BGP configuration mode.

router bgp *as-number*

(5) (Optional) Save the original routing information of a specified BGP peer.

neighbor { *neighbor-ipv4-address* | *neighbor-ipv6-address* | *peer-group-name* } **soft-reconfiguration inbound**

No original routing information of a specified BGP peer is saved by default.

1.3.10 Configuring Route Update Mechanisms of BGP

1. Overview

BGP provides two route update mechanisms: regular scanning update and event triggering update.

- Regular scanning update

  BGP uses an internal timer to start scanning regularly and update the routing table.

- Event triggering update

  When BGP configuration commands are changed due to user configuration or the next hop of a BGP route changes, BGP is triggered to start the scanning mechanism and update the routing table.

  The function of triggering routing table update by the change of the next hop of BGP is a method of reducing the BGP convergence time. This function optimizes the method of monitoring the next hops of routes, to enable BGP to speed up route convergence when the network topology is stable. When BGP establishes a connection with a neighbor, it automatically monitors the next hops of the routes learned from the neighbor. When a next hop changes in the core routing table, BGP receives a next hop change advertisement and updates the BGP routing table. This optimization measure improves the convergence performance of BGP routes by reducing the time for detecting next-hop changes.

2. Restrictions and Guidelines

- Based on address families, this function is configured in the IPv4, IPv6, IPv4 VRF, and IPv6 VRF address family modes.
- If you set the BGP route update mechanism to event-triggering update, you must disable synchronization (by running the **no synchronization** command) and enable the BGP next-hop triggering update function. Otherwise, you must not configure such update.
- The **bgp nexthop trigger delay** command and the **bgp scan-time** command control the same timer. If the BGP route update mechanism is configured as regular scanning update and the timer specified in the **bgp nexthop trigger delay** command is set to greater than 60 seconds, this command does not take effect, because the scanning timer is triggered prior to the delay of the next hop triggering update.
- In an unstable network environment with frequent hop changes, especially with many routes, the configuration of the next hop triggering update causes unnecessary route calculations, and thus consumes more CPU resources. Therefore, you are advised to disable this function in this environment.

3. Procedure

(1) Enter the privileged EXEC mode.

enable

(2) Enter the global configuration mode.

configure terminal

(3) Enable BGP to enter the BGP configuration mode.

router bgp *as-number*

(4) Configure an interval of regular scanning update of BGP.

bgp scan-time *scan-time*

The default interval of regular scanning update of BGP is **60** seconds.

(5) (Optional) Enter the address family configuration mode. To configure this function for a specific address family, please configure only one task.

- Enter the IPv4 unicast configuration mode.

  address-family ipv4 [**unicast**]

- Enter the IPv4 VRF configuration mode.

  address-family ipv4 vrf *vrf-name*

- Enter the IPv6 unicast configuration mode.

  address-family ipv6 [**unicast**]

- Enter the IPv6 VRF configuration mode.

  address-family ipv6 vrf *vrf-name*

(6) Configure the BGP route update mechanism as event triggering update. Run the following commands in turn:

- Configure the event triggering update mechanism to update the routing table.

  bgp scan-rib disable

  BGP updates the routing table in regular scanning mode by default.

- Enable the next hop triggering update.

  bgp nexthop trigger enable

  The function of next hop triggering update is enabled by default.

- Configure a delay of updating the routing table before the next hop of a BGP route changes.

  bgp nexthop trigger delay *delay-time*

  The default delay of updating the routing table is **5** seconds after the next hop of a BGP route changes.

1.3.11 Configuring BGP Capacity Protection

1. Overview

Excessive BGP routing entries can overload the device capacity, especially on devices with a small memory. BGP capacity protection can help avoid non-predictable running status caused by excessive resource consumption of the device.

- Restricting the number of BGP routing entries

  The number of BGP routing entries is restricted by configuring the maximum number of routing entries in a BGP address family and the maximum number of routing entries that are learned by a BGP neighbor.

- Enabling BGP to enter the overflow state in case of memory insufficiency

  BGP is allowed to enter the overflow state when the memory is insufficient. In the overflow state, BGP generates a default route pointing to the null interface. If a newly learned route is not a refined route of a non-default route in the current routing table, the route is discarded. The system memory is kept stable by

discarding eligible routes. The purpose of only discarding routes meeting specific conditions is to prevent route loops in the network. Therefore, BGP in the overflow state is secure.

2. Procedure

(1) Enter the privileged EXEC mode.

enable

(2) Enter the global configuration mode.

configure terminal

(3) Enable BGP to enter the BGP configuration mode.

router bgp *as-number*

(4) Configure BGP to enter the overflow state when the memory is insufficient.

overflow memory-lack

BGP enters the overflow state by default when the memory is insufficient.

(5) Enable the function of route entry limit in global configuration mode or for specified VRF instances.

bgp maximum-prefix *maximum-prefix-numbers* [**vrf** *vrf-name*]

The function of route entry limit is disabled in global configuration mode or for specified VRF instances by default.

(6) Configure the maximum number of prefixes received from specified BGP peers.

neighbor { *neighbor-ipv4-address* | *neighbor-ipv6-address* | *peer-group-name* } **maximum-prefix** *maximum-prefix-value* [*maximum-prefixthreshold*] [**restart-time** *restart-time* | **warning-only** [**suppress**]]

The number of prefixes received from a specified BGP peer is not limited by default.

(7) (Optional) Enter the address family configuration mode. To configure this function for a specific address family, please configure only one task.

- Enter the IPv4 unicast configuration mode.

  address-family ipv4 [**unicast**]

- Enter the IPv4 VRF configuration mode.

  address-family ipv4 vrf *vrf-name*

- Enter the IPv6 unicast configuration mode.

  address-family ipv6 [**unicast**]

- Enter the IPv6 VRF configuration mode.

  address-family ipv6 vrf *vrf-name*

(8) Configure the maximum number of prefixes in a BGP routing information base under the address family.

maximum-prefix *maximum-prefix-number*

The number of route prefixes in a BGP routing information base under an address family is not limited by default.

1.4 Configuring a large BGP Network

1.4.1 Configuration Tasks

The large BGP network configuration includes the following tasks: All configuration tasks are optional and may be selected as needed.

- Configuring a Route Reflector
- Configuring an AS Alliance
- Configuring Route Aggregation
- Configuring Route Dampening

1.4.2 Configuring a Route Reflector

1. Overview

A route reflector can be configured in an IBGP environment so that routing information is reflected between clients and the number of BGP neighbor connections is reduced.

2. Restrictions and Guidelines

- If you configure a route reflector in an address family, the reflector takes effect only in this address family.
- To facilitate configuration, you need to configure the reflector device. The client devices do not need to be configured as clients.
- When multiple reflection clusters are deployed in one AS domain, different reflection cluster identifiers must be configured for them.
- You can run the **exit-address-family** command to exit the address family configuration mode of BGP.

3. Procedure

(1) Enter the privileged EXEC mode.

enable

(2) Enter the global configuration mode.

configure terminal

(3) Enable BGP to enter the BGP configuration mode.

router bgp *as-number*

(4) Create a BGP neighbor.

neighbor { *neighbor-ipv4-address* | *neighbor-ipv6-address* | *peer-group-name* } **remote-as** *as-number*

(5) Configure the local device as a route reflector. Then, the BGP neighbor becomes the route reflection client.

neighbor { *neighbor-ipv4-address* | *neighbor-ipv6-address* | *peer-group-name* } **route-reflector-client**

(6) (Optional) Enable the route reflection function between clients of the devices.

bgp client-to-client reflection

The route reflection function between clients is enabled by default.

(7) (Optional) Configure a BGP reflection cluster ID.

bgp cluster-id *cluster-id*

A BGP reflection cluster ID is the BGP router-ID by default.

(8) (Optional) Allow the BGP route reflector to modify route attributes.

bgp route-reflector attribute-change

No route reflector is allowed to modify route attributes by default.

According to RFC4456, to prevent route loops, a BGP route reflector cannot modify route attributes when reflecting routes. However, relevant route attributes need to be modified when you re-plan network traffic paths. Users can configure a command to modify route attributes (including Route-map and Nexthop-self) on the route reflector.

1.4.3 Configuring an AS Alliance

1. Overview

On a large BGP network, you can configure a BGP alliance to divide an AS into multiple member ASs, to reduce the number of IBGP neighbor connections and facilitate AS management.

2. Restrictions and Guidelines

- You are advised to use private AS numbers for member ASs within an alliance. A private AS number is in the range from 64512 to 65535.
- Within a member AS of an alliance, BGP speakers must be connected in a full mesh manner or configured as reflectors and reflector clients.
- EBGP neighbor relationships must be established between member ASs of an alliance.
- All BGP speakers in an alliance must associate with a member AS of the alliance.
- All member ASs that have established EBGP relationships with the local device must be configured on members of a BGP alliance.

3. Procedure

(1) Enter the privileged EXEC mode.

enable

(2) Enter the global configuration mode.

configure terminal

(3) Enable BGP to enter the BGP configuration mode.

router bgp *as-number*

(4) Configure a BGP alliance ID. The local BGP AS (specified by the **router bgp** *as-number* command) becomes the private AS inside the alliance and is invisible to other AS domains.

bgp confederation identifier *as-number*

No alliance ID is configured for a BGP speaker by default.

(5) Configure members for the BGP alliance. The specified AS and local AS belong to the same alliance.

bgp confederation peers *as-number*&<1-n>

No alliance member is configured for BGP by default.

1.4.4 Configuring Route Aggregation

1. Overview

One or more refined BGP routes are aggregated as a BGP route with a shorter network mask.

2. Restrictions and Guidelines

- By default, BGP advertises path information before and after aggregation. To enable BGP to advertise only path information after aggregation, run the **aggregate-address summary-only** command.
- When the **aggregate-address** command is used to configure route aggregation, the aggregated routes take effect immediately as long as any route falls into the configured address range.

3. Procedure

(1) Enter the privileged EXEC mode.

enable

(2) Enter the global configuration mode.

configure terminal

(3) Enable BGP to enter the BGP configuration mode.

router bgp *as-number*

(4) Configure the route aggregation entries of BGP.

aggregate-address { *ipv4-address mask* | *prefix* } [**advertise-map** route-map-name | **as-set** | **attribute-map** *route-map-name* | **summary-only** | **suppress-map** route-map-name] *

No route aggregation entry of BGP is configured by default.

1.4.5 Configuring Route Dampening

1. Overview

BGP route dampening is a method of reducing route flapping. It reduces possible route flapping by monitoring routing information from EBGP peers. The BGP speaker punishes a route once (added to the penalty) route flapping occurs. When the penalty reaches the upper limit, the route is suppressed. When the Half-life-time reaches, the penalty is halved. When the penalty is reduced to the Reuse Limit, the route is reactivated.

2. Procedure

3. Procedure

(1) Enter the privileged EXEC mode.

enable

(2) Enter the global configuration mode.

configure terminal

(3) Enable BGP to enter the BGP configuration mode.

router bgp *as-number*

(4) Enable the route dampening function and configure dampening parameters.

bgp dampening [*half-life* [*reusing suppressing maximun-supress-time*] [**withdrawal-ignore**] | **route-map** *route-map-name*]

The route dampening function is disabled by default.

1.5 Configuring BGP Route Selection and Load Balancing

1.5.1 Configuration Tasks

The configuration of BGP route selection and load balancing includes the following tasks: All configuration tasks are optional and may be selected as needed.

- Configuring Multi-path Load Balancing of BGP
- Configuring BGP Routes to Be Recursive Only to Host Routes
- Configuring Outbound Loop Detection for a BGP Neighbor
- Configuring BGP ADD-PATH
- Configuring BGP to Advertise Routes with the Lowest Priority upon Device Restart
- Configuring the AS_PATH Attribute

1.5.2 Configuring Multi-path Load Balancing of BGP

1. Overview

This section describes how to implement multi-path load balancing for IBGP routes.

2. Restrictions and Guidelines

- Routes learned from an IBGP neighbor must have the same priority (the router-ID is not compared).
- This command does not allow IBGP and EBGP routes to form equivalent routes.
- All next hops must carry the Link-Bandwidth attribute so that non-equal-cost routes are generated. Otherwise, non-equal-cost load balancing cannot be implemented.
- Enable the function of sending extended community attributes so that the link bandwidth attribute can be sent to an IBGP neighbor.
- The function of carrying the link bandwidth attribute when a route from a specified neighbor is sent to an IBGP neighbor takes effect only on single-hop EBGP neighbors.
- You can run the **exit-address-family** command to exit the address family configuration mode of BGP.

3. Procedure

(1) Enter the privileged EXEC mode.

enable

(2) Enter the global configuration mode.

configure terminal

(3) Enable BGP to enter the BGP configuration mode.

router bgp *as-number*

(4) (Optional) Enter a required address family configuration mode. If you need to perform configuration for a specific address family, select one of the following tasks to configure.

- Enter the IPv4 unicast configuration mode.

  address-family ipv4 [**unicast**]

- Enter the IPv4 VRF configuration mode.

  address-family ipv4 vrf *vrf-name*

- Enter the IPv6 unicast configuration mode.

  address-family ipv6 [**unicast**]

- Enter the IPv6 VRF configuration mode.

  address-family ipv6 vrf *vrf-name*

(5) Create a BGP neighbor.

neighbor { *neighbor-ipv4-address* | *neighbor-ipv6-address* | *peer-group-name* } **remote-as** *as-number*

(6) Configure BGP load balancing. To configure equal-cost route load balancing by EBGP multiple paths of an alliance or by local inter-VRF imported routes, run the **maximum-paths ebgp** command.

maximum-paths { **ebgp** | **ibgp** } *maximum-paths-number*

BGP load balancing is disabled by default.

(7) (Optional) Configure the AS-PATH loose comparison mode, in which only the AS-PATH length is compared.

bgp bestpath as-path multipath-relax

The AS-PATH exact comparison mode is used by default, in which the AS-PATH attribute must be exactly the same when BGP calculates multiple equal-cost paths.

(8) (Optional) Enable the function of sending extended community attributes, to ensure that the link bandwidth attribute is transferred between neighbors.

neighbor { *neighbor-ipv4-address* | *neighbor-ipv6-address* | *peer-group-name* } **send-community** [**both** | **standard** | **extended**]

The function of advertising community attributes to a specified BGP neighbor is not configured by default.

(9) (Optional) Enable non-equal-cost load balancing. Non-equal-cost routes are generated based on the link bandwidth attribute.

bgp dmzlink-bw

Non-equal-cost load balancing is disabled by default.

(10) (Optional) Enable the function of carrying the link bandwidth attribute when a route from a specified neighbor is sent to an IBGP neighbor.

neighbor { *peer-address* | *peer-group-name* } **dmzlink-bw**

The function of carrying the link bandwidth attribute in a route from a specified neighbor is disabled by default.

1.5.3 Configuring BGP Routes to Be Recursive Only to Host Routes

1. Overview

BGP routes use optimal matching for route recursion by default. BGP routes may be recursive to default routes or incorrect network segment routes, resulting in an error in the egress or next hop. After the function of making BGP routes recursive only to host routes is enabled, BGP routes are recursive only to 32-bit IPv4 host

routes or 128-bit IPv6 host routes. This function is applied to routes learned by IBGP neighbors or multi-hop EBGP neighbors that have established neighbor relationships by using the loopback interface.

After the function of making BGP routes recursive only to host routes is enabled, the device checks the validity of the next-hop addresses of IBGP or multi-hop EBGP routes via exact matching, and conducts route recursion on the IBGP or multi-hop EBGP routes via exact matching when they are written into the forward information database (FIB). For example, the next-hop address of the IBGP route 192.168.2.0/24 is 1.1.1.1. The next-hop address of the IBGP route 192.168.2.0/24 is valid only when the route 1.1.1.1/32 exists, and the egress of the route written into the FIB is the egress of the route 1.1.1.1/32.

2. Restrictions and Guidelines

- This function takes effect only on IBGP or multi-hop EBGP routes. It is unavailable to direct EBGP routes or routes whose next-hop addresses are direct connection addresses.
- IBGP or multi-hop EBGP neighbor relationships of the device need to be established using the address of the loopback interface. Otherwise, this function cannot be enabled.

3. Procedure

(1) Enter the privileged EXEC mode.

enable

(2) Enter the global configuration mode.

configure terminal

(3) Enable BGP to enter the BGP configuration mode.

router bgp *as-number*

(4) Configure BGP routes to be recursive only to host routes.

bgp recursion host

BGP routes recurse via optimal matching by default.

1.5.4 Configuring Outbound Loop Detection for a BGP Neighbor

1. Overview

When receiving a BGP route from a neighbor, BGP conducts loop detection on the BGP route by default. If the AS-PATH attribute carried in a BGP route contains the local AS number, BGP filters out the BGP route. The BGP outbound loop detection is to conduct loop detection on routes before the routes are transmitted to a neighbor, so as to filter out loop routes.

When sending a route to an EBGP neighbor, the device judges whether the AS-PATH attribute carried in the BGP route contains the AS number of the neighbor. If yes, the route is looped and the device does not send the route to the EBGP neighbor.

2. Restrictions and Guidelines

This feature is available only to EBGP neighbors.

3. Procedure

(1) Enter the privileged EXEC mode.

enable

(2) Enter the global configuration mode.

configure terminal

(3) Enable BGP to enter the BGP configuration mode.

router bgp *as-number*

(4) Create a BGP neighbor.

neighbor { *neighbor-ipv4-address* | *neighbor-ipv6-address* | *peer-group-name* } **remote-as** *as-number*

(5) Configure outbound loop detection for the BGP neighbor

neighbor { *neighbor-ipv4-address* | *neighbor-ipv6-address* | *peer-group-name* } **as-loop-check out**

The outbound loop detection is disabled for a neighbor by default.

1.5.5 Configuring BGP ADD-PATH

1. Overview

The BGP additional paths (ADD-PATH) attribute is used to receive multiple routes with the same prefix from an IBGP neighbor. When receiving routes with the same prefix from the same peer, BGP replaces a previous route with the latest new one by default, that is, BGP regards the routes with the same prefix received from the same neighbor as one entry. After the BGP ADD-PATH function is configured, BGP adds a path-ID field to the route advertised to a neighbor that supports the ADD-PATH capability. The device judges whether routes are the same based on the route prefix and path-ID of a neighbor. When a received route is different from previous routes, the device adds it as a new entry. In this way, the device can obtain multiple routes with the same prefix from the same neighbor.

2. Restrictions and Guidelines

Run the **bgp additional-paths select** command to select ADD-PATH alternative routes of a specific type. If the type of selected ADD-PATH alternative routes is different from that of to-be-advertised ones, the ADD-PATH alternative routes are not advertised but only the optimal routes are advertised.

3. Procedure

(1) Enter the privileged EXEC mode.

enable

(2) Enter the global configuration mode.

configure terminal

(3) Enable BGP to enter the BGP configuration mode.

router bgp *as-number*

(4) Enable the ADD-PATH selection function.

bgp additional-paths select { **all** | **best** *number* | **ecmp** }

(5) Create a BGP neighbor.

neighbor { neighbor-ipv4-address | neighbor-ipv6-address | peer-group-name } **remote-as** as-number

(6) The ADD-PATH function is disabled by default. Configure the device to advertise the ADD-PATH alternative routes of a specific type to a peer when the ADD-PATH negotiation is successful.

neighbor { *peer-address* | *peer-group-name* } **advertise additional-paths** { **all** | **best** *number* | **ecmp** }

BGP does not advertise ADD-PATH alternative routes to its neighbors by default.

(7) Enable the ADD-PATH capability for a specified peer.

neighbor { *peer-address* | *peer-group-name* } **additional-paths** { **send [receive] | receive** }

The ADD-PATH capability of a peer is disabled by default.

1.5.6 Configuring BGP to Advertise Routes with the Lowest Priority upon Device Restart

1. Overview

By default, after a neighbor relationship is established upon device restart, a BGP peer can advertise routes to its neighbors. However, in certain cases, for example, a device has many neighbors, there are considerable routes during device startup, but writing entries into the hardware is slow. The neighbors have learned the routes and started forwarding traffic, but the local device has not completed the writing of entries into the hardware, causing a traffic forwarding failure. Therefore, it is necessary to adjust the priority of routes advertised by BGP to the lowest priority after the device restart. In this way, routes passing through the local device are not selected while other routes are available.

This function enables BGP to advertise routes with the lowest priority to neighbors after the local device is started and has established a neighbor relationship. For EBGP neighbors, the MED value of routes is adjusted to **4294967295**. For IBGP neighbors, the LOCAL_PREF value of routes is adjusted to **0**. After confirming that ARP entries have been delivered and forwarding entries have been written into hardware, run the **clear bgp advertise lowest-priority on-startup** command to restore the priority of advertised routes. If you do not configure the priority restoration for advertised routes, the device automatically restores the priority of advertised routes after a period of time.

2. Procedure

(1) Enter the privileged EXEC mode.

enable

(2) Enter the global configuration mode.

configure terminal

(3) Enable BGP to enter the BGP configuration mode.

router bgp *as-number*

(4) Configure BGP to adjust the priority of advertised routes to the lowest level upon device restart.

bgp advertise lowest-priority on-startup [*recover-time*]

BGP does not modify the priority of advertised routes by default.

1.5.7 Configuring the AS_PATH Attribute

1. Overview

AS_PATH Lists the ASs passed by a route reversely. Routes are selected based on different AS_PATH attributes.

2. Procedure

(1) Enter the privileged EXEC mode.

enable

(2) Enter the global configuration mode.

configure terminal

(3) Enable BGP to enter the BGP configuration mode.

router bgp *as-number*

(4) (Optional) Enter the address family configuration mode. To configure this function for a specific address family, please configure only one task.

- Enter the IPv4 unicast configuration mode.

  address-family ipv4 [**unicast**]

- Enter the IPv4 VRF configuration mode.

  address-family ipv4 vrf *vrf-name*

- Enter the IPv6 unicast configuration mode.

  address-family ipv6 [**unicast**]

- Enter the IPv6 VRF configuration mode.

  address-family ipv6 vrf *vrf-name*

(5) Create a BGP neighbor.

neighbor { *neighbor-ipv4-address* | *neighbor-ipv6-address* | *peer-group-name* } **remote-as** *as-number*

(6) Configure the local AS for the BGP neighbor.

neighbor { *neighbor-ipv4-address* | *neighbor-ipv6-address* | *peer-group-name* } **local-as** *as-number* [**no-prepend** [**replace-as** [**dual-as**]]]

No local AS is configured for any peer by default, and the local AS of a peer is the real AS of BGP.

After the local AS function is configured, when the real AS of the local BGP changes, you do not need to change the BGP configuration on the peer and a BGP connection can be still established. This function is used for AS migration and merging on large networks and ensures that the device configurations in other interconnected ASs are not affected.

1.6 Configuring BGP to Control Route Advertisement and Receiving

1.6.1 Configuration Tasks

The configuration of route advertisement and receiving control of BGP includes the following tasks: All configuration tasks are optional and may be selected as needed.

- Configuring Fast Withdrawal of Specified BGP Routes
- Configuring Delayed Route Advertisement of BGP

1.6.2 Configuring Fast Withdrawal of Specified BGP Routes

1. Overview

With the rapid development of IP technologies and the application of various complex services, higher requirements are posed for the network. Customers may need fast convergence of specific routes. This function can meet customers' requirements for preferential withdrawal of specific routes.

The route withdrawal sequence is calculated based on the routes in the routing table by default. This function adds routes that meet conditions to a high-priority queue for preferential processing, so as to achieve the purpose of rapidly withdrawing specified routes.

2. Procedure

(1) Enter the privileged EXEC mode.

enable

(2) Enter the global configuration mode.

configure terminal

(3) Enable BGP to enter the BGP configuration mode.

router bgp *as-number*

(4) (Optional) Enter the address family configuration mode. To configure this function for a specific address family, please configure only one task.

- Enter the IPv4 unicast configuration mode.

  address-family ipv4 [**unicast**]

- Enter the IPv4 VRF configuration mode.

  address-family ipv4 vrf *vrf-name*

- Enter the IPv6 unicast configuration mode.

  address-family ipv6 [**unicast**]

- Enter the IPv6 VRF configuration mode.

  address-family ipv6 vrf *vrf-name*

(5) Create a BGP neighbor.

neighbor { *neighbor-ipv4-address* | *neighbor-ipv6-address* | *peer-group-name* } **remote-as** *as-number*

(6) Configure BGP to rapidly withdraw specified routes.

bgp fast-withdraw { **access-list** { *access-list-number* | *access-list-name* } | **prefix-list** *prefix-list-name* | **route-map** *route-map-name* }

The rapid withdrawal of specified routes is disabled by default.

1.6.3 Configuring Delayed Route Advertisement of BGP

1. Overview

- Delayed advertisement upon system restart

  After the system is restarted and a neighbor relationship is established, a BGP peer can advertise routes to its neighbors. However, in certain cases, for example, a device has many neighbors or routes and the device is restarted. So, the device is slow in writing entries to hardware. As a result, the neighbors have learned the routes and started forwarding traffic, but the local device hardware has not completed writing of entries, causing a traffic forwarding failure.

  The BGP delayed advertisement upon system restart enables the device not to immediately advertise routes to its neighbors but wait for a period of time after the system is restarted and the BGP neighbor relationships are established. This function does not affect other behaviors of neighbors such as route

receiving. If some routes need to be kept from impact of the system delay, you can configure a prefix-list policy to match these routes, providing a flexible route advertisement method.

Delay-time specifies the waiting time before routes are advertised to neighbors and startup-time specifies the startup-time. Within the startup-time, BGP sends received routing information to the neighbors at an interval specified by delay-time. After the startup-time ends, the default route advertisement behavior recovers.

- Delayed first route advertisement

  Once the neighbors of BGP negotiate to reach the established status, they send update packets each other. If delayed first route advertisement of BGP is configured on the local device, the local device will send the route received from the specified neighbor to other neighbors after the delayed period.

2. Restrictions and Guidelines

- If delayed advertisement of BGP upon system restart and delayed first route advertisement of BGP are enabled at the same time, delayed advertisement of BGP upon system restart takes precedence over delayed first route advertisement for first routes of BGP.
- BGP GR is not affected by either delayed advertisement of BGP upon system restart or delayed first route advertisement of BGP, that is, the BGP GR route advertisement is not affected by the delay time.

3. Procedure

(1) Enter the privileged EXEC mode.

enable

(2) Enter the global configuration mode.

configure terminal

(3) Enable BGP to enter the BGP configuration mode.

router bgp *as-number*

(4) Enable the function of delayed advertisement of BGP upon system restart.

bgp initial-advertise-delay { *delay-time* [*startup-time*] [**wait-for-controller**] | **prefix-list** *prefix-list-name* }

The function of delayed advertisement of BGP upon system restart is disabled by default.

1.7 Configuring the BGP Security Function

1.7.1 Overview

The configuration of the BGP security function includes the following tasks: All configurations are optional and may be selected as needed.

- Configuring MD5 Authentication
- Configuring the Generalized TTL Security Mechanism (GTSM) Security Check for BGP Neighbors

1.7.2 Configuring MD5 Authentication

1. Overview

Message-digest algorithm 5 (MD5) authentication passwords are used when BGP speakers establish a TCP connection. If the authentication fails, no TCP connection will be established.

2. Restrictions and Guidelines

- The same password must be set for the BGP speakers that need to establish a BGP connection. Otherwise, no neighbor relationship can be established between them.
- After the MD5 authentication command is configured, BGP speakers re-establishes a BGP connection.
- Only one authentication password can be configured for one neighbor.
- You can run the **exit-address-family** command to exit the address family configuration mode of BGP.

3. Procedure

(1) Enter the privileged EXEC mode.

enable

(2) Enter the global configuration mode.

configure terminal

(3) Enable BGP to enter the BGP configuration mode.

router bgp *as-number*

(4) (Optional) Enter the address family configuration mode. To configure this function for a specific address family, please configure only one task.

- Enter the IPv4 unicast configuration mode.

  address-family ipv4 [**unicast**]

- Enter the IPv4 VRF configuration mode.

  address-family ipv4 vrf *vrf-name*

- Enter the IPv6 unicast configuration mode.

  address-family ipv6 [**unicast**]

- Enter the IPv6 VRF configuration mode.

  address-family ipv6 vrf *vrf-name*

(5) Create a BGP neighbor.

neighbor { *neighbor-ipv4-address* | *neighbor-ipv6-address* | *peer-group-name* } **remote-as** *as-number*

(6) Configure MD5 encryption for a TCP connection to be established with the BGP neighbor.

neighbor { *neighbor-ipv4-address* | *neighbor-ipv6-address* | *peer-group-name* } **password** [**0** | **7**] *password-string*

By default, a BGP connection is not encrypted through MD5.

1.7.3 Configuring the Generalized TTL Security Mechanism (GTSM) Security Check for BGP Neighbors

1. Overview

Network devices consume CPU resources when processing IP packets on the control plane. An attacker continuously sends false BGP packets to a device. If the device finds the packets sent to the local device upon receipt of them and forwards the packets to the control plane for processing without identification, the device consumes many CPU resources, because it needs to process these packets.

Configuring BGP GTSM security check can protect devices from CPU type attacks. GTSM security check determines whether the TTL value in IP packet headers is within the specified range. If not, the packets are invalid and directly discarded.

2. Procedure

(1) Enter the privileged EXEC mode.

enable

(2) Enter the global configuration mode.

configure terminal

(3) Enable BGP to enter the BGP configuration mode.

router bgp *as-number*

(4) Configure the function of GTSM security check for BGP neighbors.

neighbor { *neighbor-ipv4-address* | *neighbor-ipv6-address* | *peer-group-name* } **ttl-security hops** *hop-count*

The function of GTSM security check is not enabled by default.

1.8 Configuring BGP Reliability

1.8.1 Configuration Tasks

BGP reliability configuration includes the following tasks: All configurations are optional and may be selected as needed.

- Configuring BFD For BGP
- Configuring BGP FRR
- Configuring EBGP Multi-Path Bypass Protection
- Configuring BGP GR
- Configuring BGP NSR
- Configuring BGP Session Retention

1.8.2 Configuring BFD For BGP

1. Overview

You can configure bidirectional forwarding detection (BFD) for BGP to rapidly detect a network failure and switch data to a backup link, thereby improving the network use experience.

2. Restrictions and Guidelines

You can run the **exit-address-family** command to exit the address family configuration mode of BGP.

3. Prerequisites

Before configuring BFD for BGP, ensure that BFD session parameters have been configured for a relevant interface of a neighbor.

4. Procedure

(1) Enter the privileged EXEC mode.

enable

(2) Enter the global configuration mode.

configure terminal

(3) Enable BGP to enter the BGP configuration mode.

router bgp *as-number*

(4) (Optional) Enter the address family configuration mode. To configure this function for a specific address family, please configure only one task.

- Enter the IPv4 unicast configuration mode.

  address-family ipv4 [**unicast**]

- Enter the IPv4 VRF configuration mode.

  address-family ipv4 vrf *vrf-name*

- Enter the IPv6 unicast configuration mode.

  address-family ipv6 [**unicast**]

- Enter the IPv6 VRF configuration mode.

  address-family ipv6 vrf *vrf-name*

(5) Create a BGP neighbor.

neighbor { *neighbor-ipv4-address* | *neighbor-ipv6-address* | *peer-group-name* } **remote-as** *as-number*

(6) Configure BFD for BGP to detect the neighbor changes.

neighbor { *neighbor-ipv4-address* | *neighbor-ipv6-address* | *peer-group-name* } **fall-over bfd**

The BFD function is disabled by default.

1.8.3 Configuring BGP FRR

1. Overview

This section describes how to use a backup route to replace a failed route to rectify the fault locally.

2. Prerequisites

Configure a BFD session for a neighbor to implement fast link fault detection.

3. Procedure

(1) Enter the privileged EXEC mode.

enable

(2) Enter the global configuration mode.

configure terminal

(3) Enable BGP to enter the BGP configuration mode.

router bgp *as-number*

(4) (Optional) Enter the address family configuration mode. To configure this function for a specific address family, please configure only one task.

- Enter the IPv4 unicast configuration mode.

  address-family ipv4 [**unicast**]

- Enter the IPv4 VRF configuration mode.

  address-family ipv4 vrf *vrf-name*

- Enter the IPv6 unicast configuration mode.

  address-family ipv6 [**unicast**]

- Enter the IPv6 VRF configuration mode.

  address-family ipv6 vrf *vrf-name*

(5) Create a BGP neighbor.

neighbor { *neighbor-ipv4-address* | *neighbor-ipv6-address* | *peer-group-name* } **remote-as** *as-number*

(6) Configure BGP FRR.

bgp fast-reroute

The FRR function is disabled by default.

1.8.4 Configuring EBGP Multi-Path Bypass Protection

1. Overview

After EBGP multi-path bypass protection is configured, when all ECMP routes fail, the bypass route becomes the primary route for data forwarding.

BGP FRR supports only 1:1 protection. After the equal-cost multi-path routing (ECMP) function is enabled, FRR becomes unavailable. The multi-path bypass protection enables BGP to support 1:N protection. Users can configure BGP multi-path bypass protection to enable the system to select a backup bypass path even if ECMP is configured. When all ECMP routes fail, the bypass route becomes the primary route for forwarding.

When there are multiple paths to the same network in the routing table of BGP, BGP calculates a route with the highest priority by default. After the BGP multi-path bypass protection function is enabled, BGP selects a preferred backup route even if ECMP routes exist. After the fast link detection mechanism of BFD detects that the master link is faulty, the system switches data to the originally calculated backup link for forwarding. After route convergence is completed, the device switches data to the re-calculated optimal path for data forwarding.

2. Restrictions and Guidelines

You can configure a neighbor BFD session to implement fast link fault detection.

3. Procedure

(1) Enter the privileged EXEC mode.

enable

(2) Enter the global configuration mode.

configure terminal

(3) Enable BGP to enter the BGP configuration mode.

router bgp *as-number*

(4) Configure BGP multi-path bypass protection.

bgp install standby-path

Multi-path bypass protection is disabled by default.

(5) Create a BGP neighbor.

neighbor { *neighbor-ipv4-address* | *neighbor-ipv6-address* | *peer-group-name* } **remote-as** *as-number*

(6) Create a BFD session for the BGP neighbor.

neighbor { *neighbor-ipv4-address* | *neighbor-ipv6-address* | *peer-group-name* } **fall-over bfd**

No BFD session is configured for a BGP neighbor by default.

1.8.5 Configuring BGP GR

1. Overview

This section describes how to configure BGP GR to ensure that data transmission is not interrupted during protocol restart and implement the deployment of highly reliable networks.

2. Restrictions and Guidelines

- To successfully deploy the BGP GR function, you need to configure a neighbor as the GR Helper.
- IGP GR also needs to be configured in an IBGP environment.
- After BGP GR is enabled, you need to reset a BGP neighbor connection to make BGP GR take effect.
- After the GR function is enabled in BGP configuration mode, the GR function will be enabled in all address families.
- You can run the **exit-address-family** command to exit the address family configuration mode of BGP.

3. Procedure

(1) Enter the privileged EXEC mode.

enable

(2) Enter the global configuration mode.

configure terminal

(3) Enable BGP to enter the BGP configuration mode.

router bgp *as-number*

(4) Create a BGP neighbor.

neighbor { *neighbor-ipv4-address* | *neighbor-ipv6-address* | *peer-group-name* } **remote-as** *as-number*

(5) Configure BGP GR.

bgp graceful-restart

The GR function is enabled by default.

(6) Configure the BGP GR restart timer.

bgp graceful-restart restart-time *restart-time*

The default timer value is **120** seconds.

(7) (Optional) Configure the BGP GR routing aging timer.

bgp graceful-restart stalepath-time *time*

The default value of the aging timer is **360** seconds.

1.8.6 Configuring BGP NSR

1. Overview

During switchover between master and slave supervisor engines, the NSR function keeps the network topology stable, maintains neighbor status and forwarding table, and ensures that key services are not interrupted.

2. Restrictions and Guidelines

- To deploy the BGP NSR function, rely on the device to support the function of dual engine redundancy.
- After the BGP NSR function is enabled, connections to the BGP neighbors will be reset for the function to take effect.
- The neighbor based NSR command is equivalent to the **bgp nsr** command. This function enables the NSR function for all neighbors.
- You are not advised to use the neighbor-based NSR command as it is to be abandoned.
- When the BGP NSR function is enabled for a neighbor, the connection to the neighbor will be reset and the TCP none-stop-service (NSS) will be enabled for the neighbor. Then, relevant neighbor and routing information will be backed up to the slave supervisor engine. In this case, if the NSR function is enabled after a BGP speaker is in the established state, the neighbor relationship will be re-established, causing neighbor flapping. You are advised to enable NSR before a neighbor relationship is established between peers.
- The NSR function does not take effect on a neighbor that fails to establish a BGP connection. That is, if the BGP connection of a neighbor is not in the established state, the neighbor will not execute the NSR operation. Only a neighbor in the established state will perform NSR-related backup operations.
- Supporting the BGP NSR function does not mean that the device can perform switchovers between the active and standby management boards. To perform such switchovers, you must depend on the device hardware. NSR devices must support the function of dual engine redundancy.
- If the BGP or TCP backup is not complete, the current BGP NSR switchover may fail. When a NSR switchover failure occurs, routes can be still recovered through BGP GR. Therefore, you are advised to enable both BGP NSR and BGP GR.

3. Procedure

(1) Enter the privileged EXEC mode.

enable

(2) Enter the global configuration mode.

configure terminal

(3) Enable BGP to enter the BGP configuration mode.

router bgp *as-number*

(4) Create a BGP neighbor.

neighbor { *neighbor-ipv4-address* | *neighbor-ipv6-address* | *peer-group-name* } **remote-as** *as-number*

(5) Configure the BGP NSR function.

bgp nsr

The NSR function is disabled by default.

1.8.7 Configuring BGP Session Retention

1. Overview

For devices that support BGP, routing errors in an address family will affect route stability in other address families: When a device receives an update packet from a neighbor and an error on the multi-protocol routing attribute is detected, the device will disconnect the BGP session. As a result, flapping occurs on routes in all address families of this neighbor.

After the BGP session retention function is enabled, if an error occurs in the routing attribute of an address family, only the routing information related to the neighbor in this address family is deleted. The BGP session and other address families are not affected, which enhances the stability of BGP.

Recovery-time specifies the waiting time for automatic route recovery. A neighbor must support the route-refresh capability so that the time can be set. After the recovery-time elapses, BGP sends the route-refresh message of the address family to the neighbor and re-advertises all routing information in the address family to this neighbor.

2. Procedure

(1) Enter the privileged EXEC mode.

enable

(2) Enter the global configuration mode.

configure terminal

(3) Enable BGP to enter the BGP configuration mode.

router bgp *as-number*

(4) Configure BGP to retain BGP sessions when it detects a multiprotocol path error.

bgp mp-error-handle session-retain [**refresh-timer** *refresh-timer*]

BGP terminates BGP sessions by default when it detects a multiprotocol route error.

1.9 Enabling the Extended Functions of BGP

1.9.1 Configuration Tasks

The configuration of BGP extended functions includes the following tasks: All configuration tasks are optional and may be selected as needed.

- Configuring BMP Monitoring
- Configuring the Administrative Distance of BGP

1.9.2 Configuring BMP Monitoring

1. Overview

After BMP monitoring is configured, BMP is monitored by the BMP server, which focuses on the status of BGP peers, peer entry transmission and receiving, and all kinds of peer statistics.

2. Procedure

(1) Enter the privileged EXEC mode.

enable

(2) Enter the global configuration mode.

configure terminal

(3) Create a BMP server instance.

bmp server *server-number*

(4) Configure the address and port number of the BMP server.

address [*bmp-server-ipv4-address* | *bmp-server-ipv6-address*] **port** *port-number*

The address and port number of the BMP server are not configured in BMP mode by default.

(5) Configure an outbound interface for establishing a TCP connection with the BMP server.

update-source *interface-type interface-number*

The system automatically selects the outbound interface for a TCP connection based on the IP address of the BMP server by default. The outbound interface of local packets is generally used.

(6) Configure the VRF instance, to which a TCP connection to the BMP server belongs.

vrf *vrf-name*

A TCP connection is established with the BMP server in the default VRF instance by default.

(7) (Optional) Configure the interval for re-establishing a TCP connection with the BMP server.

failure-retry-delay *time*

The interval for reconnecting to the BMP server gradually increases from the initial **30** seconds by default.

(8) (Optional) Configure a description for the BMP server instance.

description *text*

No description is configured for a BMP server instance by default.

(9) Configure the BMP server to monitor routes that are received from peers and have a routing policy applied.

adj-rib-in post-policy

The BMP server does not monitor routes that are received from a BGP peer and have a routing policy applied by default.

(10) Configure the BMP server to monitor routes that are received from a peer and are not renamed.

adj-rib-in pre-policy

The BMP server does not monitor routes that are received from a BGP peer and are not renamed by default.

(11) Configure the BMP server to monitor routes that are sent to a peer and have a routing policy applied.

adj-rib-out post-policy

The BMP server does not monitor routes that are sent to a BGP peer and have a routing policy applied by default.

(12) Configure the BMP server to monitor a BGP peer.

neighbor { *neighbor-ipv4-address* | *neighbor-ipv6-address* | *peer-group-name* } **bmp-active** { **all** | **server** *server-number* [*server-number* [*server-number*]] }

BGP peers are not monitored by the BMP server by default.

(13) Enable the packet mirroring function of BGP.

route mirroring

The BMP server does not send route mirroring packets by default.

(14) Configure the interval for BMP server to send status statistics.

stats-reporting-period *time*

When status statistics of a monitored peer are changed, BMP delays a period of time in sending the status statistics to the BMP server by default.

1.9.3 Configuring the Administrative Distance of BGP

1. Overview

The administrative distance is used to evaluate the reliability of various route sources. A smaller administrative distance indicates a better route.

- Administrative distance of BGP

  The administrative distance indicates the reliability of a route source. The value range is from 1 to 255. A larger administrative distance indicates a lower route reliability. BGP sets different administrative distances for sources, from which routing information is learned. Administrative distances are classified into external-distances, internal-distances, and local-distances.

  - External-distance: Indicates the administrative distances of routes learned from EBGP peers.
  - Internal-distance: Indicates the administrative distances of routes learned from IBGP peers.
  - Local-distance: Indicates the administrative distances of routes learned from peers but it is considered that better routes can be learned from IGP. Generally, these routes can be indicated by the **network backdoor** command.

- Backdoor route

  If you prefer an IGP route without using this EBGP route, you can configure this EBGP route as the backdoor route. The default administrative distance of routes learned from a BGP speaker that has established an EBGP connection is **20**. You can run the **network backdoor** command to set the administrative distance of EBGP routes to 200 so that routes learned from IGP have a higher priority. Routes learned from IGP are considered as backdoor routes and are not advertised externally.

2. Procedure

(1) Enter the privileged EXEC mode.

enable

(2) Enter the global configuration mode.

configure terminal

(3) Enable BGP to enter the BGP configuration mode.

router bgp *as-number*

(4) Configure the administrative distance of a BGP route.

distance bgp *external-distance internal-distance local-distance*

The default administrative distance of a route learned by BGP from an EBGP peer is **20** and that of a route learned by BGP from an IBGP peer is **200**.

(5) Configure a backdoor route.

network *network-number* [**mask** *mask*] [**route-map** *map-tag*] **backdoor**

No backdoor route is configured by default.

1.10 Monitoring

Run the **show** commands to check the running status of a configured function to verify the configuration effect.

Run the **clear** commands to clear information.

⚠ Caution

Running the **clear** command during operation of the device may lose vital information and interrupt services.

Run the **debug** commands to output debugging information.

⚠ Caution

The output debugging information occupies system resources. Therefore, disable the debugging switch immediately after use.

Table 1-1 Monitoring

Command	Purpose
clear ip bgp [**vrf** *vrf-name*] { * | *as-number* | *neighbor-ipv4-address* | *neighbor-ipv6-address* } [**soft**] [**in** | **out**] **clear bgp ipv4 unicast** [**vrf** *vrf-name*] { ***** | *as-number* | *neighbor-ipv4-address* | *neighbor-ipv6-address* } [**soft**] [**in** | **out**] **clear ip bgp** [**vrf** *vrf-name*] **update-group** [*update-group-index* | *neighbor-ipv4-address* | *neighbor-ipv6-address*] [**soft**] [**in** | **out**] **clear bgp ipv4 unicast** [**vrf** *vrf-name*] **update-group** [*update-group-index* | *neighbor-ipv4-address* | *neighbor-ipv6-address*] [**soft**] [**in** | **out**]	Clears BGP IPv4 unicast routes.

Command	Purpose
clear bgp ipv6 unicast [**vrf** *vrf-name*] { ***** | *as-number* | *neighbor-ipv4-address* | *neighbor-ipv6-address* } [**soft**] [**in** | **out**] **clear bgp ipv6 unicast** [**vrf** *vrf-name*] **update-group** [*update-group-index* | *neighbor-ipv4-address* | *neighbor-ipv6-address*] [**soft**] [**in** | **out**]	Clears BGP IPv6 unicast routes.
clear bgp advertise lowest-priority on-startup	Restores the priority of routes advertised by BGP to a neighbor to the one before the priority is adjusted to the lowest.
show ip bgp **show bgp ipv4 unicast**	Displays BGP IPv4 unicast routes.
show ip bgp [**vrf** *vrf-name*] **update-group** [*neighbor-ipv4-address* | *neighbor-ipv6-address* | *update-group-index*] [**summary**] **show bgp ipv4 unicast** [**vrf** *vrf-name*] **update-group** [*neighbor-ipv4-address* | *neighbor-ipv6-address* | *update-group-index*] [**summary**]	Displays the update-group information of the BGP IPv4 unicast address family.
show bgp ipv6 unicast	Displays BGP IPv6 unicast routes.
show bgp ipv6 unicast [**vrf** *vrf-name*] **update-group** [*neighbor-ipv4-address* | *neighbor-ipv6-address* | *update-group-index*] [**summary**]	Displays the update-group information of the BGP IPv6 unicast address family.
show bgp bmp server [*server-number* [**detail**] | **detail**] }	Displays information about a BMP server instance.
show bgp bmp neighbor	Displays information about neighbors monitored by the BMP server.
show bgp bmp summary	Displays the summary of an established connection to the BMP server.
show bgp statistics [**vrf** *vrf-name*]	Displays BGP statistics.
debug ip bgp all	Debugs BGP.
debug ip bgp bmp	Debugs BGP BMP.
debug ip bgp dampening	Debugs route flapping of BGP.
debug ip bgp event	Debugs event processing of BGP.
debug ip bgp filter	Debugs route filtering of BGP.
debug ip bgp fsm	Debugs BGP state machine.

Command	Purpose
debug ip bgp keepalives	Debugs neighbor keepalive of BGP.
debug ip bgp nsm	Debugs core route processing of BGP.
debug ip bgp update	Debugs BGP update packets.
debug ip bgp update-group	Debugs the update-group function of BGP.
debug ip bgp add-path	Debugs the ADD-PATH function of BGP.

1.11 Configuration Examples

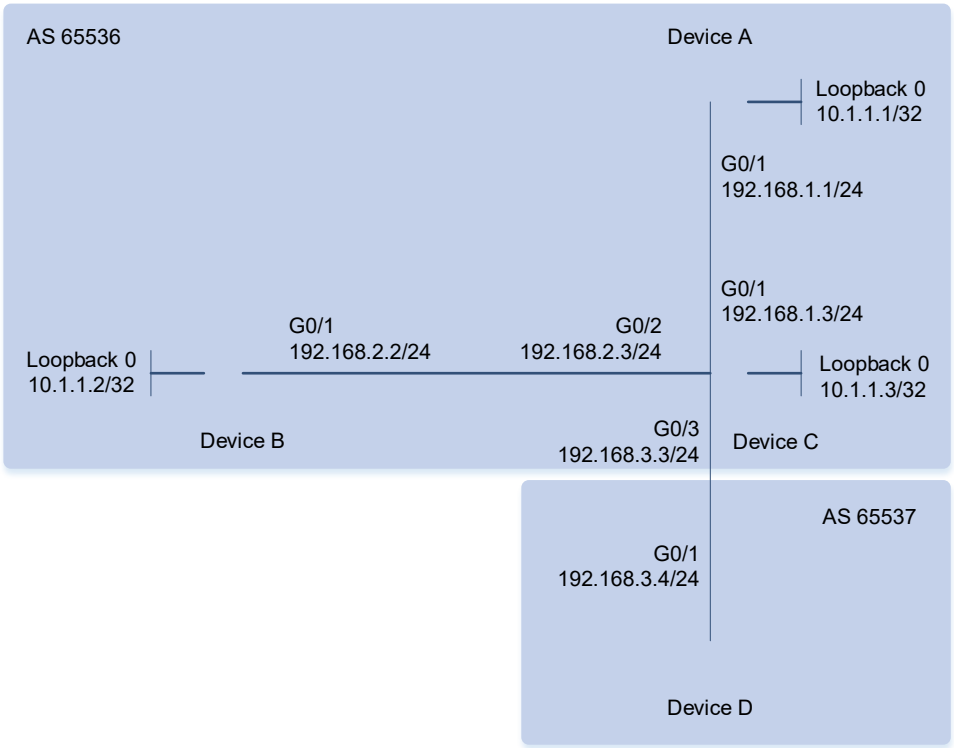
1.11.1 Configuring BGP Basic Networking

1. Requirements

BGP speakers need to establish IBGP neighbor relationships and EBGP neighbor relationships to implement basic networking.

2. Topology

Figure 1-1 Topology of BGP Basic Networking



3. Notes

- Enable BGP on all devices and set AS numbers for the devices as shown in Figure 1-1 (omitted).

- Configure a loopback interface on each of devices A, B, and C, which establish IBGP neighbor relationships through the loopback interfaces.
- Configure devices C and D to establish an EBGP neighbor relationship through directly connected interfaces.
- Create an IBGP peer group on device C.

4. Procedure

(1) Configure loopback interfaces and IP addresses for the interfaces.

Configure device A.

```
Device A# configure terminal
Device A(config)# interface loopback 0
Device A(config-if-Loopback 0)# ip address 10.1.1.1 255.255.255.255
Device A(config-if-Loopback 0)# exit
Device A(config)# interface GigabitEthernet 0/1
Device A(config-if-GigabitEthernet 0/1)# ip address 192.168.1.1 255.255.255.0
Device A(config-if-GigabitEthernet 0/1)# exit
```

Configure device B.

```
Device B# configure terminal
Device B(config)# interface loopback 0
Device B(config-if-Loopback 0)# ip address 10.1.1.2 255.255.255.255
Device B(config-if-Loopback 0)# exit
Device B(config)# interface GigabitEthernet 0/1
Device B(config-if-GigabitEthernet 0/1)# ip address 192.168.2.2 255.255.255.0
Device B(config-if-GigabitEthernet 0/1)# exit
```

Configure device C.

```
Device C# configure terminal
Device C(config)# interface loopback 0
Device C(config-if-Loopback 0)# ip address 10.1.1.3 255.255.255.255
Device C(config-if-Loopback 0)# exit
Device C(config)# interface GigabitEthernet 0/1
Device C(config-if-GigabitEthernet 0/1)# ip address 192.168.1.3 255.255.255.0
Device C(config-if-GigabitEthernet 0/1)# exit
Device C(config)# interface GigabitEthernet 0/2
Device C(config-if-GigabitEthernet 0/2)# ip address 192.168.2.3 255.255.255.0
Device C(config-if-GigabitEthernet 0/2)# exit
Device C(config)# interface GigabitEthernet 0/3
Device C(config-if-GigabitEthernet 0/3)# ip address 192.168.3.3 255.255.255.0
Device C(config-if-GigabitEthernet 0/3)# exit
```

Configure device D.

```
Device D# configure terminal
Device D(config)# interface GigabitEthernet 0/1
Device D(config-if-GigabitEthernet 0/1)# ip address 192.168.3.4 255.255.255.0
Device D(config-if-GigabitEthernet 0/1)# exit
```

(2) Enable BGP and configure the loopback interfaces as neighbor interfaces.

Configure device A.

```
Device A(config)# router bgp 65536
Device A(config-router)# neighbor 10.1.1.3 remote-as 65536
Device A(config-router)# neighbor 10.1.1.3 update-source loopback 0
```

Configure device B.

```
Device B(config)# router bgp 65536
Device B(config-router)# neighbor 10.1.1.3 remote-as 65536
Device B(config-router)# neighbor 10.1.1.3 update-source loopback 0
```

Configure device C.

```
Device C(config)# router bgp 65536
Device C(config-router)# neighbor ibgp-group peer-group
Device C(config-router)# neighbor ibgp-group remote-as 65536
Device C(config-router)# neighbor ibgp-group update-source loopback 0
Device C(config-router)# neighbor 10.1.1.1 peer-group ibgp-group
Device C(config-router)# neighbor 10.1.1.2 peer-group ibgp-group
Device C(config-router)# neighbor 192.168.3.4 remote-as 65537
```

Configure device D.

```
Device D(config)# router bgp 65537
Device D(config-router)# neighbor 192.168.3.3 remote-as 65536
```

5. Verification

On device A, run the **show ip bgp neighbor** command to display the BGP neighbor status.

```
Device A# show ip bgp neighbor
BGP neighbor is 10.1.1.3, remote AS 65536, local AS 65536, internal link
  BGP version 4, remote router ID 10.1.1.3
  BGP state = Established, up for 00:00:05
  Last read         , hold time is 180, keepalive interval is 60 seconds
  Neighbor capabilities:
    Route refresh: advertised and received (old and new)
    Four-octets ASN Capability: advertised and received
    Address family IPv4 Unicast: advertised and received
  Received 2 messages, 0 notifications, 0 in queue
    open message:1 update message:0 keepalive message:1
    refresh message:0 dynamic cap:0 notifications:0
  Sent 2 messages, 0 notifications, 0 in queue
    open message:1 update message:0 keepalive message:1
    refresh message:0 dynamic cap:0 notifications:0
  Route refresh request: received 0, sent 0
  Minimum time between advertisement runs is 0 seconds
  Update source is Loopback 0

 For address family: IPv4 Unicast
  BGP table version 1, neighbor version 1
  Index 0, Offset 0, Mask 0x1
```

```
  0 accepted prefixes
  0 announced prefixes

 Connections established 1; dropped 0
Local host: 10.1.1.1, Local port: 1039
Foreign host: 10.1.1.3, Foreign port: 179
Nexthop: 10.1.1.1
Nexthop global: ::
Nexthop local: ::
BGP connection: non shared network
Last Reset:          , due to BGP Notification received
Notification Error Message: (Cease/Other Configuration Change.)
```

On device B, run the **show** command to display the BGP neighbor status.

```
Device B# show ip bgp neighbor
BGP neighbor is 10.1.1.3, remote AS 65536, local AS 65536, internal link
  BGP version 4, remote router ID 10.1.1.3
  BGP state = Established, up for 00:00:07
  Last read          , hold time is 180, keepalive interval is 60 seconds
  Neighbor capabilities:
    Route refresh: advertised and received (old and new)
    Four-octets ASN Capability: advertised and received
    Address family IPv4 Unicast: advertised and received
  Received 2 messages, 0 notifications, 0 in queue
    open message:1 update message:0 keepalive message:1
    refresh message:0 dynamic cap:0 notifications:0
  Sent 2 messages, 0 notifications, 0 in queue
    open message:1 update message:0 keepalive message:1
    refresh message:0 dynamic cap:0 notifications:0
  Route refresh request: received 0, sent 0
  Minimum time between advertisement runs is 0 seconds
  Update source is Loopback 0

 For address family: IPv4 Unicast
  BGP table version 1, neighbor version 1
  Index 0, Offset 0, Mask 0x1
  0 accepted prefixes
  0 announced prefixes

 Connections established 1; dropped 0
Local host: 10.1.1.2, Local port: 1041
Foreign host: 10.1.1.3, Foreign port: 179
Nexthop: 10.1.1.2
Nexthop global: ::
Nexthop local: ::
BGP connection: non shared network
Last Reset:          , due to BGP Notification received
```

```
Notification Error Message: (Cease/Other Configuration Change.)
```

On device C, run the **show** command to display the BGP neighbor status.

```
Device C# show ip bgp neighbor
BGP neighbor is 10.1.1.1, remote AS 65536, local AS 65536, internal link
 Member of peer-group ibgp-group for session parameters
  BGP version 4, remote router ID 10.1.1.1
  BGP state = Established, up for 00:01:13
  Last read         , hold time is 180, keepalive interval is 60 seconds
  Neighbor capabilities:
    Route refresh: advertised and received (old and new)
    Four-octets ASN Capability: advertised and received
    Address family IPv4 Unicast: advertised and received
  Received 3 messages, 0 notifications, 0 in queue
    open message:1 update message:0 keepalive message:2
    refresh message:0 dynamic cap:0 notifications:0
  Sent 3 messages, 0 notifications, 0 in queue
    open message:1 update message:0 keepalive message:2
    refresh message:0 dynamic cap:0 notifications:0
  Route refresh request: received 0, sent 0
  Minimum time between advertisement runs is 0 seconds
  Update source is Loopback 0

 For address family: IPv4 Unicast
  BGP table version 1, neighbor version 1
  Index 1, Offset 0, Mask 0x2
  ibgp-group peer-group member
  0 accepted prefixes
  0 announced prefixes

 Connections established 1; dropped 0
Local host: 10.1.1.3, Local port: 179
Foreign host: 10.1.1.1, Foreign port: 1039
Nexthop: 10.1.1.3
Nexthop global: ::
Nexthop local: ::
BGP connection: non shared network

BGP neighbor is 10.1.1.2, remote AS 65536, local AS 65536, internal link
 Member of peer-group ibgp-group for session parameters
  BGP version 4, remote router ID 10.1.1.2
  BGP state = Established, up for 00:01:17
  Last read         , hold time is 180, keepalive interval is 60 seconds
  Neighbor capabilities:
    Route refresh: advertised and received (old and new)
    Four-octets ASN Capability: advertised and received
    Address family IPv4 Unicast: advertised and received
```

```
  Received 3 messages, 0 notifications, 0 in queue
    open message:1 update message:0 keepalive message:2
    refresh message:0 dynamic cap:0 notifications:0
  Sent 3 messages, 0 notifications, 0 in queue
    open message:1 update message:0 keepalive message:2
    refresh message:0 dynamic cap:0 notifications:0
  Route refresh request: received 0, sent 0
  Minimum time between advertisement runs is 0 seconds
  Update source is Loopback 0

 For address family: IPv4 Unicast
  BGP table version 1, neighbor version 1
  Index 1, Offset 0, Mask 0x2
  ibgp-group peer-group member
  0 accepted prefixes
  0 announced prefixes

 Connections established 1; dropped 0
Local host: 10.1.1.3, Local port: 179
Foreign host: 10.1.1.2, Foreign port: 1041
Nexthop: 10.1.1.3
Nexthop global: ::
Nexthop local: ::
BGP connection: non shared network

BGP neighbor is 192.168.3.4, remote AS 65536, local AS 65536, internal link
 Member of peer-group ibgp-group for session parameters
  BGP version 4, remote router ID 192.168.3.4
  BGP state = Established, up for 00:01:01
  Last read         , hold time is 180, keepalive interval is 60 seconds
  Neighbor capabilities:
    Route refresh: advertised and received (old and new)
    Four-octets ASN Capability: advertised and received
    Address family IPv4 Unicast: advertised and received
  Received 3 messages, 0 notifications, 0 in queue
    open message:1 update message:0 keepalive message:2
    refresh message:0 dynamic cap:0 notifications:0
  Sent 3 messages, 0 notifications, 0 in queue
    open message:1 update message:0 keepalive message:2
    refresh message:0 dynamic cap:0 notifications:0
  Route refresh request: received 0, sent 0
  Minimum time between advertisement runs is 0 seconds
  Update source is Loopback 0

 For address family: IPv4 Unicast
  BGP table version 1, neighbor version 1
```

```
  Index 1, Offset 0, Mask 0x2
  ibgp-group peer-group member
  0 accepted prefixes
  0 announced prefixes

 Connections established 1; dropped 0
Local host: 192.168.3.3, Local port: 179
Foreign host: 192.168.3.4, Foreign port: 1018
Nexthop: 192.168.3.3
Nexthop global: ::
Nexthop local: ::
BGP connection: non shared network
```

On device D, run the **show** command to display the BGP neighbor status.

```
Device D# show ip bgp neighbor
BGP neighbor is 192.168.3.3, remote AS 65536, local AS 65536, internal link
 Member of peer-group ibgp-group for session parameters
  BGP version 4, remote router ID 10.1.1.3
  BGP state = Established, up for 00:01:01
  Last read         , hold time is 180, keepalive interval is 60 seconds
  Neighbor capabilities:
    Route refresh: advertised and received (old and new)
    Four-octets ASN Capability: advertised and received
    Address family IPv4 Unicast: advertised and received
  Received 3 messages, 0 notifications, 0 in queue
    open message:1 update message:0 keepalive message:2
    refresh message:0 dynamic cap:0 notifications:0
  Sent 3 messages, 0 notifications, 0 in queue
    open message:1 update message:0 keepalive message:2
    refresh message:0 dynamic cap:0 notifications:0
  Route refresh request: received 0, sent 0
  Minimum time between advertisement runs is 0 seconds
  Update source is Loopback 0

 For address family: IPv4 Unicast
  BGP table version 1, neighbor version 1
  Index 1, Offset 0, Mask 0x2
  ibgp-group peer-group member
  0 accepted prefixes
  0 announced prefixes

 Connections established 1; dropped 0
Local host: 192.168.3.4, Local port: 1018
Foreign host: 192.168.3.3, Foreign port: 179
Nexthop: 192.168.3.4
Nexthop global: ::
Nexthop local: ::
```

```
BGP connection: non shared network
```

6. Configuration Files

- Device A configuration file

```
!
interface GigabitEthernet 0/1
 no switchport
 ip address 192.168.1.1 255.255.255.0
!
interface Loopback 0
 ip address 10.1.1.1 255.255.255.255
!
router bgp 65536
 bgp log-neighbor-changes
 bgp graceful-restart restart-time 120
 bgp graceful-restart stalepath-time 360
 bgp graceful-restart
 neighbor 10.1.1.3 remote-as 65536
 neighbor 10.1.1.3 update-source Loopback 0
 address-family ipv4
  neighbor 10.1.1.3 activate
  exit-address-family
!
```

- Device B configuration file

```
!
interface GigabitEthernet 0/1
 no switchport
 ip address 192.168.2.2 255.255.255.0
!
interface Loopback 0
 ip address 10.1.1.2 255.255.255.255
!
router bgp 65536
 bgp log-neighbor-changes
 bgp graceful-restart restart-time 120
 bgp graceful-restart stalepath-time 360
 bgp graceful-restart
 neighbor 10.1.1.3 remote-as 65536
 neighbor 10.1.1.3 update-source Loopback 0
 address-family ipv4
  neighbor 10.1.1.3 activate
  exit-address-family
!
```

- Device C configuration file

```
!
```

```
interface GigabitEthernet 0/1
 no switchport
 ip address 192.168.1.3 255.255.255.0
!
interface GigabitEthernet 0/2
 no switchport
 ip address 192.168.2.3 255.255.255.0
!
interface GigabitEthernet 0/3
 no switchport
 ip address 192.168.3.3 255.255.255.0
!
interface Loopback 0
 ip address 10.1.1.3 255.255.255.255
!
router bgp 65536
 bgp log-neighbor-changes
 bgp graceful-restart restart-time 120
 bgp graceful-restart stalepath-time 360
 bgp graceful-restart
 neighbor ibgp-group peer-group
 neighbor ibgp-group remote-as 65536
 neighbor ibgp-group update-source Loopback 0
 neighbor 10.1.1.1 peer-group ibgp-group
 neighbor 10.1.1.2 peer-group ibgp-group
 neighbor 192.168.3.4 remote-as 65537
 address-family ipv4
  no neighbor ibgp-group activate
  neighbor 192.168.3.4 activate
  exit-address-family
!
```

- Device D configuration file

```
!
interface GigabitEthernet 0/1
 no switchport
 ip address 192.168.1.3 255.255.255.0
!
router bgp 65537
 bgp log-neighbor-changes
 bgp graceful-restart restart-time 120
 bgp graceful-restart stalepath-time 360
 bgp graceful-restart
 address-family ipv4
  exit-address-family
!
```

7. Common Errors

- IGP is disabled and the local loopback address and the loopback address of an IBGP neighbor cannot interwork with each other, resulting in the failure to establish a neighbor relationship.
- When EBGP neighbors are not directly connected, **ebgp-multihop** is not configured, causing the failure to establish a TCP connection.

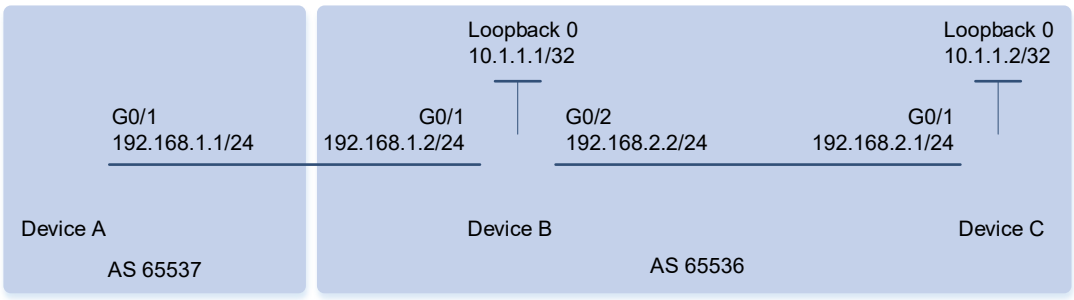
1.11.2 Configuring BGP MD5 Authentication

1. Requirements

The establishment of a TCP connection needs MD5 encryption to ensure data transmission security.

2. Topology

Figure 1-1 Topology of BGP MD5 Authentication



3. Notes

- Enable BGP on all devices and set AS numbers for the devices as shown in Figure 1-1 (omitted).
- Configure a loopback interface on each of devices B and C, which establish an IBGP neighbor relationship through the loopback interfaces (omitted).
- Configure devices A and B to establish an EBGP neighbor relationship through directly connected interfaces.
- On devices A, B, and C, configure passwords for their neighbors.

4. Procedure

Configure device A.

```
Device A# configure terminal
Device A(config)# router bgp 65537
Device A(config-router)# neighbor 192.168.1.2 remote-as 65536
Device A(config-router)# neighbor 192.168.1.2 password 7 ebgpneighbor
```

Configure device B.

```
Device B# configure terminal
Device B(config)# router bgp 65536
Device B(config-router)# neighbor 10.1.1.2 remote-as 65536
Device B(config-router)# neighbor 10.1.1.2 update-source loopback 0
Device B(config-router)# neighbor 10.1.1.2 password ibgpneighbor
Device B(config-router)# neighbor 192.168.1.1 remote-as 65537
Device B(config-router)# neighbor 192.168.1.1 password 7 ebgpneighbor
```

Configure device C.

```
Device C# configure terminal
Device C(config)# router bgp 65536
Device C(config-router)# neighbor 10.1.1.1 remote-as 65536
Device C(config-router)# neighbor 10.1.1.1 update-source loopback 0
Device C(config-router)# neighbor 10.1.1.1 password ibgpneighbor
```

5. Verification

On device A, run the **show ip bgp neighbors** command to display the BGP neighbor status.

```
Device A# show ip bgp neighbors
BGP neighbor is 192.168.1.2, remote AS 65536, local AS 65537, external link
  BGP version 4, remote router ID 10.1.1.1
  BGP state = Established, up for 00:04:54
  Last read         , hold time is 180, keepalive interval is 60 seconds
  Neighbor capabilities:
    Route refresh: advertised and received (old and new)
    Four-octets ASN Capability: advertised and received
    Address family IPv4 Unicast: advertised and received
  Received 7 messages, 0 notifications, 0 in queue
    open message:1 update message:0 keepalive message:6
    refresh message:0 dynamic cap:0 notifications:0
  Sent 7 messages, 0 notifications, 0 in queue
    open message:1 update message:0 keepalive message:6
    refresh message:0 dynamic cap:0 notifications:0
  Route refresh request: received 0, sent 0
  Minimum time between advertisement runs is 30 seconds

 For address family: IPv4 Unicast
  BGP table version 1, neighbor version 0
  Index 1, Offset 0, Mask 0x2
  0 accepted prefixes
  0 announced prefixes

 Connections established 2; dropped 1
Local host: 192.168.1.1, Local port: 1026
Foreign host: 192.168.1.2, Foreign port: 179
Nexthop: 192.168.1.1
Nexthop global: ::
Nexthop local: ::
BGP connection: non shared network
Last Reset: 00:04:54, due to BGP Notification sent
Notification Error Message: (Cease/Administratively Reset.)
```

On device B, run the **show ip bgp neighbors** command to display the BGP neighbor status.

```
Device B# show ip bgp neighbors
BGP neighbor is 10.1.1.2, remote AS 65536, local AS 65536, internal link
```

```
  BGP version 4, remote router ID 10.1.1.2
  BGP state = Established, up for 00:04:01
  Last read         , hold time is 180, keepalive interval is 60 seconds
  Neighbor capabilities:
    Route refresh: advertised and received (old and new)
    Four-octets ASN Capability: advertised and received
    Address family IPv4 Unicast: advertised and received
  Received 8 messages, 0 notifications, 0 in queue
    open message:1 update message:0 keepalive message:7
    refresh message:0 dynamic cap:0 notifications:0
  Sent 8 messages, 0 notifications, 0 in queue
    open message:1 update message:0 keepalive message:7
    refresh message:0 dynamic cap:0 notifications:0
  Route refresh request: received 0, sent 0
  Minimum time between advertisement runs is 30 seconds

 For address family: IPv4 Unicast
  BGP table version 1, neighbor version 0
  Index 1, Offset 0, Mask 0x2
  0 accepted prefixes
  0 announced prefixes

 Connections established 2; dropped 1
Local host: 10.1.1.1, Local port: 179
Foreign host: 10.1.1.2, Foreign port: 1038
Nexthop: 10.1.1.1
Nexthop global: ::
Nexthop local: ::
BGP connection: non shared network
Last Reset: 00:05:27, due to BGP Notification received
Notification Error Message: (Cease/Administratively Reset.)

BGP neighbor is 192.168.1.1, remote AS 65537, local AS 65536, external link
  BGP version 4, remote router ID 192.168.1.1
  BGP state = Established, up for 00:05:27
  Last read         , hold time is 180, keepalive interval is 60 seconds
  Neighbor capabilities:
    Route refresh: advertised and received (old and new)
    Four-octets ASN Capability: advertised and received
    Address family IPv4 Unicast: advertised and received
  Received 8 messages, 0 notifications, 0 in queue
    open message:1 update message:0 keepalive message:7
    refresh message:0 dynamic cap:0 notifications:0
  Sent 8 messages, 0 notifications, 0 in queue
    open message:1 update message:0 keepalive message:7
    refresh message:0 dynamic cap:0 notifications:0
```

```
  Route refresh request: received 0, sent 0
  Minimum time between advertisement runs is 30 seconds

 For address family: IPv4 Unicast
  BGP table version 1, neighbor version 0
  Index 1, Offset 0, Mask 0x2
  0 accepted prefixes
  0 announced prefixes

 Connections established 2; dropped 1
Local host: 192.168.1.2, Local port: 179
Foreign host: 192.168.1.1, Foreign port: 1026
Nexthop: 192.168.1.2
Nexthop global: ::
Nexthop local: ::
BGP connection: non shared network
Last Reset: 00:05:27, due to BGP Notification received
Notification Error Message: (Cease/Administratively Reset.)
```

On device C, run the **show ip bgp neighbors** command to display the BGP neighbor status.

```
Device C# show ip bgp neighbors
BGP neighbor is 10.1.1.1, remote AS 65536, local AS 65536, internal link
  BGP version 4, remote router ID 10.1.1.1
  BGP state = Established, up for 00:04:01
  Last read         , hold time is 180, keepalive interval is 60 seconds
  Neighbor capabilities:
    Route refresh: advertised and received (old and new)
    Four-octets ASN Capability: advertised and received
    Address family IPv4 Unicast: advertised and received
  Received 8 messages, 0 notifications, 0 in queue
    open message:1 update message:0 keepalive message:7
    refresh message:0 dynamic cap:0 notifications:0
  Sent 8 messages, 0 notifications, 0 in queue
    open message:1 update message:0 keepalive message:7
    refresh message:0 dynamic cap:0 notifications:0
  Route refresh request: received 0, sent 0
  Minimum time between advertisement runs is 30 seconds

 For address family: IPv4 Unicast
  BGP table version 1, neighbor version 0
  Index 1, Offset 0, Mask 0x2
  0 accepted prefixes
  0 announced prefixes

 Connections established 2; dropped 1
Local host: 10.1.1.2, Local port: 1038
Foreign host: 10.1.1.1, Foreign port: 179
```

```
Nexthop: 10.1.1.2
Nexthop global: ::
Nexthop local: ::
BGP connection: non shared network
Last Reset: 00:05:27, due to BGP Notification received
Notification Error Message: (Cease/Administratively Reset.)
```

6. Configuration Files

- Device A configuration file

```
!
router bgp 65537
 bgp log-neighbor-changes
 bgp graceful-restart restart-time 120
 bgp graceful-restart stalepath-time 360
 bgp graceful-restart
 neighbor 192.168.1.2 remote-as 65536
 neighbor 192.168.1.2 password 7 ebgpneighbor
 address-family ipv4
  neighbor 192.168.1.2 activate
  exit-address-family
!
```

- Device B configuration file

```
!
router bgp 65536
 bgp log-neighbor-changes
 bgp graceful-restart restart-time 120
 bgp graceful-restart stalepath-time 360
 bgp graceful-restart
 neighbor 10.1.1.2 remote-as 65536
 neighbor 10.1.1.2 update-source Loopback 0
 neighbor 10.1.1.2 password ibgpneighbor
 neighbor 192.168.1.1 remote-as 65537
 neighbor 192.168.1.1 password 7 ebgpneighbor
 address-family ipv4
  neighbor 10.1.1.2 activate
  neighbor 192.168.1.1 activate
  exit-address-family
!
```

- Device C configuration file

```
!
router bgp 65536
 bgp log-neighbor-changes
 bgp graceful-restart restart-time 120
 bgp graceful-restart stalepath-time 360
 bgp graceful-restart
```

```
 neighbor 10.1.1.1 remote-as 65536
 neighbor 10.1.1.1 update-source Loopback 0
 neighbor 10.1.1.1 password ibgpneighbor
 address-family ipv4
  neighbor 10.1.1.1 activate
  exit-address-family
!
```

7. Common Errors

- The passwords for MD5 authentication are different on both sides of a BGP neighbor relationship.

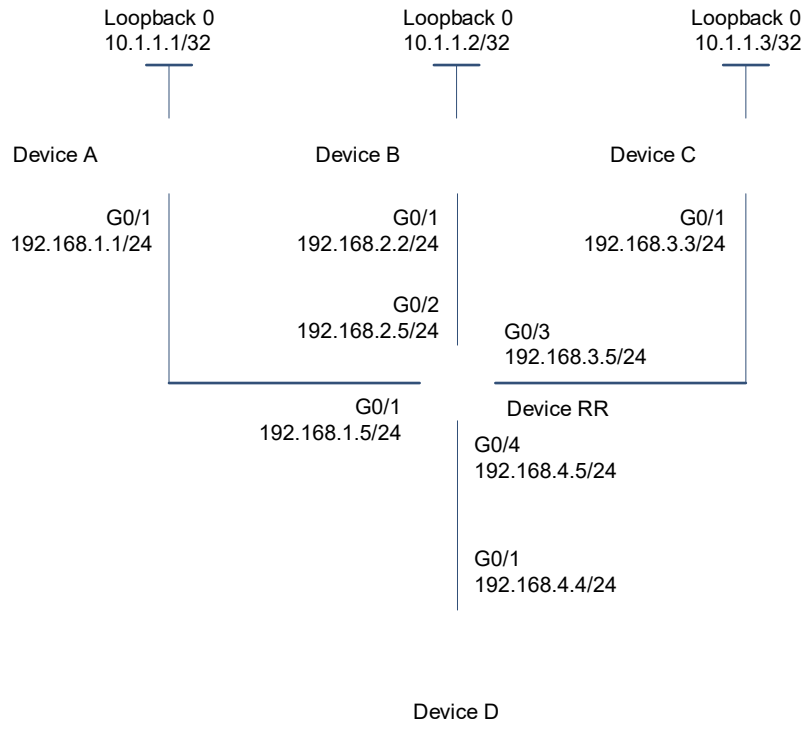
1.11.3 Configuring a BGP Route Reflector

1. Requirements

In a network involving many BGP speakers, a route reflector needs to be configured to simplify the management and synchronization of routing information and make routing more efficient.

2. Topology

Figure 1-1 Topology of BGP Route Reflector



3. Notes

- Enable BGP on all devices and set AS numbers for the devices as shown in Figure 1-1 (omitted).
- Configure a loopback interface on each device and configure the devices to establish IBGP neighbor relationships through the loopback interfaces according to connection lines shown in the figure above (omitted).
- Configure route reflection on device RR and specify devices A, B, C, and D as reflection clients.

4. Procedure

(1) Configure BGP neighbors to advertise network information.

Configure device A.

```
Device A# configure terminal
Device A(config)# router bgp 65536
Device A(config-router)# neighbor 10.1.1.5 remote-as 65536
Device A(config-router)# neighbor 10.1.1.5 update-source loopback 0
Device A(config-router)# network 192.168.1.0 mask 255.255.255.0
```

Configure device B.

```
Device B# configure terminal
Device B(config)# router bgp 65536
Device B(config-router)# neighbor 10.1.1.5 remote-as 65536
Device B(config-router)# neighbor 10.1.1.5 update-source loopback 0
```

Configure device C.

```
Device C# configure terminal
Device C(config)# router bgp 65536
Device C(config-router)# neighbor 10.1.1.5 remote-as 65536
Device C(config-router)# neighbor 10.1.1.5 update-source loopback 0
```

Configure device D.

```
Device D# configure terminal
Device D(config)# router bgp 65536
Device D(config-router)# neighbor 10.1.1.5 remote-as 65536
Device D(config-router)# neighbor 10.1.1.5 update-source loopback 0
```

(2) Configure a route reflector.

Configure device RR.

```
Device RR# configure terminal
Device RR(config)# router bgp 65536
Device RR(config-router)# neighbor 10.1.1.1 remote-as 65536
Device RR(config-router)# neighbor 10.1.1.1 update-source loopback 0
Device RR(config-router)# neighbor 10.1.1.1 route-reflector-client
Device RR(config-router)# neighbor 10.1.1.2 remote-as 65536
Device RR(config-router)# neighbor 10.1.1.2 update-source loopback 0
Device RR(config-router)# neighbor 10.1.1.2 route-reflector-client
Device RR(config-router)# neighbor 10.1.1.3 remote-as 65536
Device RR(config-router)# neighbor 10.1.1.3 update-source loopback 0
Device RR(config-router)# neighbor 10.1.1.3 route-reflector-client
Device RR(config-router)# neighbor 10.1.1.4 remote-as 65536
Device RR(config-router)# neighbor 10.1.1.4 update-source loopback 0
Device RR(config-router)# neighbor 10.1.1.4 route-reflector-client
```

5. Verification

On device RR, check the BGP neighbor status.

```
Device RR# show ip bgp summary
```

```
BGP router identifier 10.1.1.5, local AS number 65536
BGP table version is 1
0 BGP AS-PATH entries
0 BGP Community entries
1 BGP Prefix entries (Maximum-prefix:4294967295)

Neighbor        V           AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down
State/PfxRcd
10.1.1.1        4       65536       8       9        1    0    0 00:05:11
1
10.1.1.2        4       65536       9       9        1    0    0 00:05:24
0
10.1.1.3        4       65536       8       7        1    0    0 00:05:10
0
10.1.1.4        4       65536       9       8        1    0    0 00:05:14
0

Device RR# show ip bgp
BGP table version is 1, local router ID is 10.1.1.5
Status codes: s suppressed, d damped, h history, * valid, > best, i -
internal,
              S Stale, b - backup entry
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric      LocPrf      Weight Path
*>i192.168.1.0      10.1.1.1                 0         100           0    i

Total number of prefixes 1
```

On device D, run the **show ip bgp summary** command to display BGP details.

```
Device D# show ip bgp summary
BGP router identifier 10.1.1.4, local AS number 65536
BGP table version is 1
0 BGP AS-PATH entries
0 BGP Community entries
1 BGP Prefix entries (Maximum-prefix:4294967295)

Neighbor        V           AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down
State/PfxRcd
10.1.1.5        4       65536       8       9        1    0    0 00:05:20
1

Device D# show ip bgp
BGP table version is 1, local router ID is 10.1.1.4
Status codes: s suppressed, d damped, h history, * valid, > best, i -
internal,
              S Stale, b - backup entry
```

```
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric      LocPrf      Weight Path
* i192.168.1.0      10.1.1.1                 0         100           0    i

Total number of prefixes 1
```

6. Configuration Files

- Device A configuration file

```
!
router bgp 65536
 bgp log-neighbor-changes
 bgp graceful-restart restart-time 120
 bgp graceful-restart stalepath-time 360
 bgp graceful-restart
 neighbor 10.1.1.5 remote-as 65536
 neighbor 10.1.1.5 update-source Loopback 0
 address-family ipv4
  network 192.168.1.0
  neighbor 10.1.1.5 activate
  exit-address-family
!
```

- Device B configuration file

```
!
router bgp 65536
 bgp log-neighbor-changes
 bgp graceful-restart restart-time 120
 bgp graceful-restart stalepath-time 360
 bgp graceful-restart
 neighbor 10.1.1.5 remote-as 65536
 neighbor 10.1.1.5 update-source Loopback 0
 address-family ipv4
  neighbor 10.1.1.5 activate
  exit-address-family
!
```

- Device C configuration file

```
!
router bgp 65536
 bgp log-neighbor-changes
 bgp graceful-restart restart-time 120
 bgp graceful-restart stalepath-time 360
 bgp graceful-restart
 neighbor 10.1.1.5 remote-as 65536
 neighbor 10.1.1.5 update-source Loopback 0
 address-family ipv4
```

```
  neighbor 10.1.1.5 activate
  exit-address-family
!
```

- Device D configuration file

```
!
router bgp 65536
 bgp log-neighbor-changes
 bgp graceful-restart restart-time 120
 bgp graceful-restart stalepath-time 360
 bgp graceful-restart
 neighbor 10.1.1.5 remote-as 65536
 neighbor 10.1.1.5 update-source Loopback 0
 address-family ipv4
  neighbor 10.1.1.5 activate
  exit-address-family
!
```

- Device RR configuration file

```
!
router bgp 65536
 bgp confederation identifier 100
 bgp confederation peers 65537
 bgp log-neighbor-changes
 bgp graceful-restart restart-time 120
 bgp graceful-restart stalepath-time 360
 bgp graceful-restart
 neighbor 10.1.1.1 remote-as 65536
 neighbor 10.1.1.1 update-source Loopback 0
 neighbor 10.1.1.2 remote-as 65536
 neighbor 10.1.1.2 update-source Loopback 0
 neighbor 10.1.1.3 remote-as 65536
 neighbor 10.1.1.3 update-source Loopback 0
 neighbor 10.1.1.4 remote-as 65536
 neighbor 10.1.1.4 update-source Loopback 0
 address-family ipv4
  network 192.168.1.0
  neighbor 10.1.1.1 activate
  neighbor 10.1.1.1 route-reflector-client
  neighbor 10.1.1.2 activate
  neighbor 10.1.1.2 route-reflector-client
  neighbor 10.1.1.3 activate
  neighbor 10.1.1.3 route-reflector-client
  neighbor 10.1.1.4 activate
  neighbor 10.1.1.4 route-reflector-client
  exit-address-family
!
```

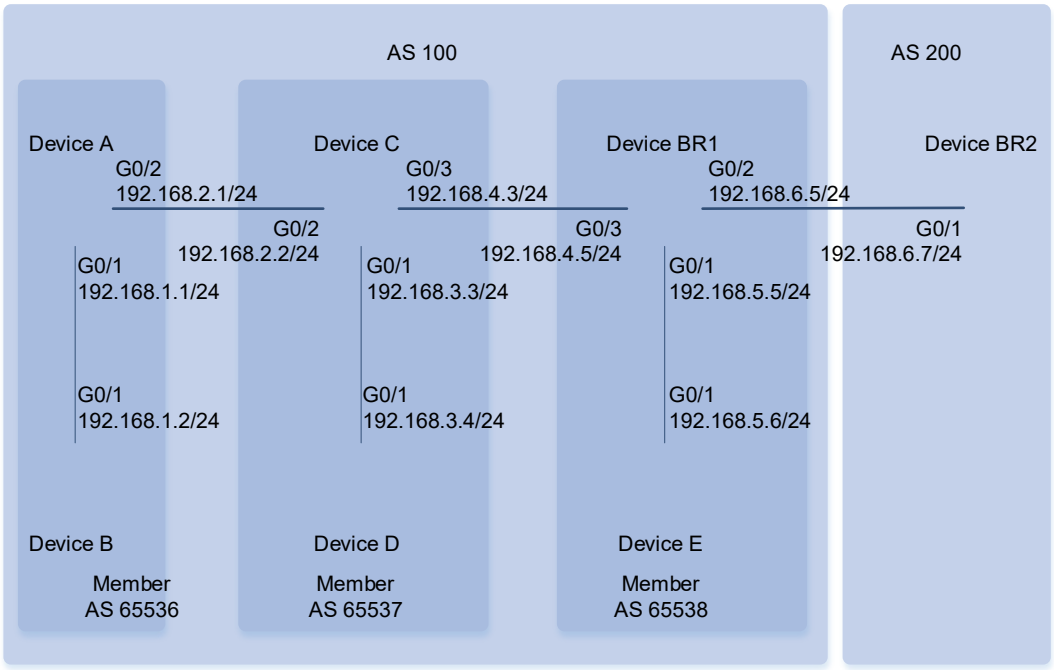
1.11.4 Configuring a BGP Alliance

1. Requirements

A BGP alliance needs to be configured to solve the problem of too many maintenance resources due to excessive IBGP connections in the network.

2. Topology

Figure 1-1 Topology of BGP Alliance



3. Notes

- Configure IP addresses for device interfaces (omitted).
- On devices A and B, configure BGP, set the member AS number to **65536**, and configure an IBGP neighbor relationship.
- On devices C and D, configure BGP, set the member AS number to **65537**, and configure an IBGP neighbor relationship.
- On devices BR1 and E, configure BGP, set the member AS number to **65538**, and configure an IBGP neighbor relationship.
- On devices A, B, C, D, E, and BR1, set the alliance ID to **100**.
- On device A, configure 65537 as the alliance number, configure device C as an EBGP neighbor, and set the peer AS number to **65537**.
- On device C, set the alliance member numbers to **65536** and **65538**, configure device A as an EBGP neighbor and set the peer AS number to **65536**, configure device BR1 as an EBGP neighbor and set the peer AS number to **65538**.
- On device BR1, set the alliance member number to **65537**, configure device C as an EBGP neighbor and set the peer AS number to **65537**, configure device BR2 as an EBGP neighbor and set the peer AS number

to **200**.

- On device BR2, configure BGP, set the AS number to **200**, configure device BR1 an EBGP neighbor, and set the peer AS number to **100**.

4. Procedure

(1) Configure IBGP neighbors and EBGP neighbors to advertise network information.

Configure device A.

```
Device A# configure terminal
Device A(config)# router bgp 65536
Device A(config-router)# neighbor 10.1.1.2 remote-as 65536
Device A(config-router)# neighbor 10.1.1.2 update-source loopback 0
Device A(config-router)# neighbor 10.1.1.3 remote-as 65537
Device A(config-router)# neighbor 10.1.1.3 ebgp-multihop 2
Device A(config-router)# neighbor 10.1.1.3 update-source loopback 0
Device A(config-router)# network 192.168.1.0 mask 255.255.255.0
```

Configure device B.

```
Device B# configure terminal
Device B(config)# router bgp 65536
Device B(config-router)# neighbor 10.1.1.1 remote-as 65536
Device B(config-router)# neighbor 10.1.1.1 update-source loopback 0
```

Configure device C.

```
Device C# configure terminal
Device C(config)# router bgp 65537
Device C(config-router)# neighbor 10.1.1.1 remote-as 65536
Device C(config-router)# neighbor 10.1.1.1 update-source loopback 0
Device C(config-router)# neighbor 10.1.1.1 ebgp-multihop 2
Device C(config-router)# neighbor 10.1.1.4 remote-as 65537
Device C(config-router)# neighbor 10.1.1.4 update-source loopback 0
Device C(config-router)# neighbor 10.1.1.5 remote-as 65538
Device C(config-router)# neighbor 10.1.1.5 update-source loopback 0
Device C(config-router)# neighbor 10.1.1.5 ebgp-multihop 2
```

Configure device D.

```
Device D# configure terminal
Device D(config)# router bgp 65537
Device D(config-router)# neighbor 10.1.1.3 remote-as 65537
Device D(config-router)# neighbor 10.1.1.3 update-source loopback 0
```

Configure device E.

```
Device E# configure terminal
Device E(config)# router bgp 65538
Device E(config-router)# neighbor 10.1.1.5 remote-as 65538
Device E(config-router)# neighbor 10.1.1.5 update-source loopback 0
```

Configure device BR1.

```
Device BR1# configure terminal
```

```
Device BR1(config)# router bgp 65538
Device BR1(config-router)# neighbor 10.1.1.3 remote-as 65537
Device BR1(config-router)# neighbor 10.1.1.3 update-source loopback 0
Device BR1(config-router)# neighbor 10.1.1.3 ebgp-multihop 2
Device BR1(config-router)# neighbor 10.1.1.6 remote-as 65538
Device BR1(config-router)# neighbor 10.1.1.6 update-source loopback 0
Device BR1(config-router)# neighbor 192.168.6.7 remote-as 200
```

Configure device BR2.

```
Device BR2# configure terminal
Device BR2(config)# router bgp 200
Device BR2(config-router)# neighbor 192.168.6.5 remote-as 100
Device BR2(config-router)# network 192.168.6.0 mask 255.255.255.0
```

(2) Configuring alliance members.

Configure device A.

```
Device A# configure terminal
Device A(config)# router bgp 65536
Device A(config-router)# bgp confederation identifier 100
Device A(config-router)# bgp confederation peers 65537
```

Configure device C.

```
Device C# configure terminal
Device C(config)# router bgp 65537
Device C(config-router)# bgp confederation identifier 100
Device C(config-router)# bgp confederation peers 65536 65538
```

Configure device BR1.

```
Device BR1# configure terminal
Device BR1(config)# router bgp 65538
Device BR1(config-router)# bgp confederation identifier 100
Device BR1(config-router)# bgp confederation peers 65537
```

5. Verification

On device A, check BGP information.

```
Device A# show ip bgp summary
BGP router identifier 10.1.1.1, local AS number 65536
BGP table version is 1
1 BGP AS-PATH entries
0 BGP Community entries
1 BGP Prefix entries (Maximum-prefix:4294967295)

Neighbor        V          AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down
State/PfxRcd
10.1.1.2        4       65536       3       3        1    0    0 00:00:05
0
10.1.1.3        4       65537       3       3        1    0    0 00:00:06
1
```

```
Total number of neighbors 1

Device A# show ip bgp
BGP table version is 1, local router ID is 10.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i -
internal,
              S Stale, b - backup entry
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric      LocPrf      Weight Path
*  192.168.6.0      192.168.6.7              0         100           0 (65537
65538) 200 i

Total number of prefixes 1
```

On device BR1, check BGP information.

```
Device BR1# show ip bgp summary
BGP router identifier 10.1.1.5, local AS number 200
BGP table version is 2
2 BGP AS-PATH entries
0 BGP Community entries
2 BGP Prefix entries (Maximum-prefix:4294967295)

Neighbor        V          AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down
State/PfxRcd
10.1.1.3        4       65537       3       3        2    0    0 00:00:10
1
10.1.1.6        4       65538       3       3        2    0    0 00:00:08
0
192.168.6.7     4         200       3       3        2    0    0 00:00:05
1

Total number of neighbors 1

Device BR1# show ip bgp
BGP table version is 1, local router ID is 10.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i -
internal,
              S Stale, b - backup entry
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric      LocPrf      Weight Path
*  192.168.1.0      10.1.1.1                 0         100           0 (65537
65536) i
*> 192.168.6.0      192.168.6.7              0         100           0 200 i
```

```
Total number of prefixes 1
```

On device BR2, check BGP information.

```
Device BR2# show ip bgp summary
BGP router identifier 192.168.6.7, local AS number 200
BGP table version is 1
1 BGP AS-PATH entries
0 BGP Community entries
1 BGP Prefix entries (Maximum-prefix:4294967295)

Neighbor        V           AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down
State/PfxRcd
192.168.6.5     4          100       3       3        1    0    0 00:00:05
1

Total number of neighbors 1

Device BR2# show ip bgp
BGP table version is 1, local router ID is 10.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i -
internal,
              S Stale, b - backup entry
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric      LocPrf      Weight Path
*> 192.168.1.0      192.168.6.5              0         100           0 (65537
65538) 200 i

Total number of prefixes 1
```

6. Configuration Files

- Device A configuration file

```
!
router bgp 65536
 bgp confederation identifier 100
 bgp confederation peers 65537
 bgp log-neighbor-changes
 bgp graceful-restart restart-time 120
 bgp graceful-restart stalepath-time 360
 bgp graceful-restart
 neighbor 10.1.1.2 remote-as 65536
 neighbor 10.1.1.2 update-source Loopback 0
 neighbor 10.1.1.3 remote-as 65537
 neighbor 10.1.1.3 ebgp-multihop 2
 neighbor 10.1.1.3 update-source Loopback 0
 address-family ipv4
  network 192.168.1.0
```

```
  neighbor 10.1.1.2 activate
  neighbor 10.1.1.3 activate
  exit-address-family
!
```

- Device B configuration file

```
!
router bgp 65536
 bgp log-neighbor-changes
 bgp graceful-restart restart-time 120
 bgp graceful-restart stalepath-time 360
 bgp graceful-restart
 neighbor 10.1.1.1 remote-as 65536
 neighbor 10.1.1.1 update-source Loopback 0
 address-family ipv4
  neighbor 10.1.1.1 activate
  exit-address-family
!
```

- Device C configuration file

```
!
router bgp 65537
 bgp confederation identifier 100
 bgp confederation peers 65536 65538
 bgp log-neighbor-changes
 bgp graceful-restart restart-time 120
 bgp graceful-restart stalepath-time 360
 bgp graceful-restart
 neighbor 10.1.1.1 remote-as 65536
 neighbor 10.1.1.1 ebgp-multihop 2
 neighbor 10.1.1.1 update-source Loopback 0
 neighbor 10.1.1.4 remote-as 65537
 neighbor 10.1.1.4 update-source Loopback 0
 neighbor 10.1.1.5 remote-as 65538
 neighbor 10.1.1.5 ebgp-multihop 2
 neighbor 10.1.1.5 update-source Loopback 0
 address-family ipv4
  neighbor 10.1.1.1 activate
  neighbor 10.1.1.4 activate
  neighbor 10.1.1.5 activate
  exit-address-family
!
```

- Device D configuration file

```
!
router bgp 65537
 bgp log-neighbor-changes
 bgp graceful-restart restart-time 120
```

```
 bgp graceful-restart stalepath-time 360
 bgp graceful-restart
 neighbor 10.1.1.3 remote-as 65537
 neighbor 10.1.1.3 update-source Loopback 0
 address-family ipv4
  neighbor 10.1.1.3 activate
  exit-address-family
!
```

- Device E configuration file

```
!
router bgp 65538
 bgp log-neighbor-changes
 bgp graceful-restart restart-time 120
 bgp graceful-restart stalepath-time 360
 bgp graceful-restart
 neighbor 10.1.1.5 remote-as 65538
 neighbor 10.1.1.5 update-source Loopback 0
 address-family ipv4
  neighbor 10.1.1.5 activate
  exit-address-family
!
```

- Device BR1 configuration file

```
!
router bgp 65538
 bgp confederation identifier 100
 bgp confederation peers 65537
 bgp log-neighbor-changes
 bgp graceful-restart restart-time 120
 bgp graceful-restart stalepath-time 360
 bgp graceful-restart
 neighbor 10.1.1.3 remote-as 65537
 neighbor 10.1.1.3 ebgp-multihop 2
 neighbor 10.1.1.3 update-source Loopback 0
 neighbor 10.1.1.6 remote-as 65538
 neighbor 10.1.1.6 update-source Loopback 0
 neighbor 192.168.6.7 remote-as 200
 address-family ipv4
  neighbor 10.1.1.3 activate
  neighbor 10.1.1.6 activate
  neighbor 192.168.6.7 activate
  exit-address-family
!
```

- Device BR2 configuration file

```
!
router bgp 200
```

```
bgp log-neighbor-changes
bgp graceful-restart restart-time 120
bgp graceful-restart stalepath-time 360
bgp graceful-restart
neighbor 192.168.6.5 remote-as 100
address-family ipv4
 network 192.168.6.0
 neighbor 192.168.6.5 activate
 exit-address-family
!
```

7. Common Errors

- No BGP alliance neighbor is configured.
- A full mesh of connections is not deployed within member ASs of an alliance.

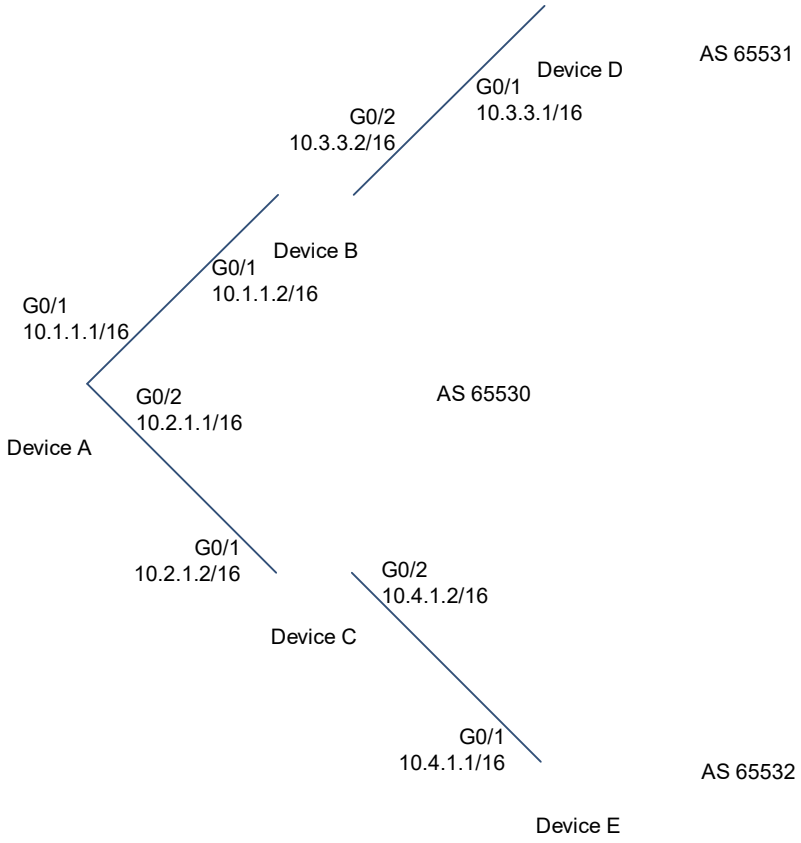
1.11.5 Configuring IBGP Multi-Path Load Balancing

1. Requirements

When there are two routes to the same destination network segment, load balancing needs to be configured to fully utilize network bandwidth resources.

2. Topology

Figure 1-1 Topology of IBGP Multi-Path Load Balancing



3. Notes

- Enable BGP on all devices and set AS numbers for the devices as shown in Figure 1-1 (omitted).
- Configure devices A and B and devices A and C to establish IBGP neighbor relationships through directly connected interfaces (omitted).
- Configure devices B and D and devices C and E to establish EBGP neighbor relationships through directly connected interfaces (omitted).
- Import the same routes to devices D and E.
- On device A, configure IBGP load balancing and enable the AS-PATH loose comparison mode.

4. Procedure

(1) Configure routing information on device A, and re-advertise directly connected route information on devices D and E.

Configure device A.

```
Device A# configure terminal
```

```
Device A(config)# ip route 10.3.0.0 255.255.0.0 10.1.1.2
Device A(config)# ip route 10.4.0.0 255.255.0.0 10.2.1.2
```

Configure device D.

```
Device D# configure terminal
Device D(config)# router bgp 65531
Device D(config-router)# redistribute connected
```

Configure device E.

```
Device E# configure terminal
Device E(config)# router bgp 65532
Device E(config-router)# redistribute connected
```

(2) On device A, configure load balancing in AS-PATH loose comparison mode.

Configure device A.

```
Device A# configure terminal
Device A(config)# router bgp 65530
Device A(config-router)# maximum-paths ibgp 2
Device A(config-router)# bgp bestpath as-path multipath-relax
```

5. Verification

On device A, check BGP information.

```
Device A# show ip bgp summary
BGP router identifier 10.2.1.1, local AS number 65530
BGP table version is 9
2 BGP AS-PATH entries
0 BGP Community entries
3 BGP Prefix entries (Maximum-prefix:4294967295)

Neighbor        V   AS      MsgRcvd   MsgSent   TblVer   InQ   OutQ   Up/Down
State/PfxRcd
172.16.23.140   4   65530   29        25        8        0     0      00:18:48
2
172.16.23.141   4   65530   24        21        8        0     0      00:17:58
2

Device A# show ip bgp
BGP table version is 9, local router ID is 10.2.1.1
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
*>i10.3.0.0/16      10.3.1.1                 0         100           0 65531 ?
*>i10.4.0.0/16      10.4.1.1                 0         100           0 65532 ?
* i10.5.0.0/16      10.3.1.1                 0         100           0 65531 ?
```

```
*>i                10.4.1.1                 0         100           0 65532 ?

Total number of prefixes 3
Device A# show ip bgp 10.5.0.0
BGP routing table entry for 10.5.0.0/16
Paths: (2 available, best #1, table Default-IP-Routing-Table)
  Not advertised to any peer
  65532
    10.4.1.1 from 10.2.1.2 (172.16.24.1)
      Origin incomplete, metric 0, localpref 100, valid, internal, multipath,
best
      Last update: Mon Mar 21 03:45:14 2011

  65531
    10.3.1.1 from 10.1.1.2 (172.16.25.1)
      Origin incomplete, metric 0, localpref 100, valid, internal, multipath
      Last update: Mon Mar 21 03:45:14 2011

Device A# show ip route

Codes:  C - connected, S - static, R - RIP, B - BGP
        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
        i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
        ia - IS-IS inter area, * - candidate default

Gateway of last resort is no set
C    10.1.0.0/16 is directly connected, FastEthernet 0/0
C    10.1.1.1/32 is local host.
C    10.2.0.0/16 is directly connected, FastEthernet 0/1
C    10.2.1.1/32 is local host.
S    10.3.0.0/16 [1/0] via 10.1.1.2
S    10.4.0.0/16 [1/0] via 10.2.1.2
B    10.5.0.0/16 [200/0] via 10.3.1.1, 00:27:56
                 [200/0] via 10.4.1.1, 00:27:56
```

6. Configuration Files

- Device A configuration file

```
!
router bgp 65530
 bgp bestpath as-path multipath-relax
 bgp log-neighbor-changes
 bgp graceful-restart restart-time 120
 bgp graceful-restart stalepath-time 360
 bgp graceful-restart
```

```
 address-family ipv4
  maximum-paths ibgp 2
  exit-address-family
!
ip route 10.3.0.0 255.255.0.0 10.1.1.2
ip route 10.4.0.0 255.255.0.0 10.2.1.2
!
```

- Device D configuration file

```
!
router bgp 65531
 bgp log-neighbor-changes
 bgp graceful-restart restart-time 120
 bgp graceful-restart stalepath-time 360
 bgp graceful-restart
 address-family ipv4
  redistribute connected
  exit-address-family
!
```

- Device E configuration file

```
!
router bgp 65532
 bgp log-neighbor-changes
 bgp graceful-restart restart-time 120
 bgp graceful-restart stalepath-time 360
 bgp graceful-restart
 address-family ipv4
  redistribute connected
  exit-address-family
!
```

7. Common Errors

The priorities of multi-hop BGP paths are different, which causes a load balancing failure.

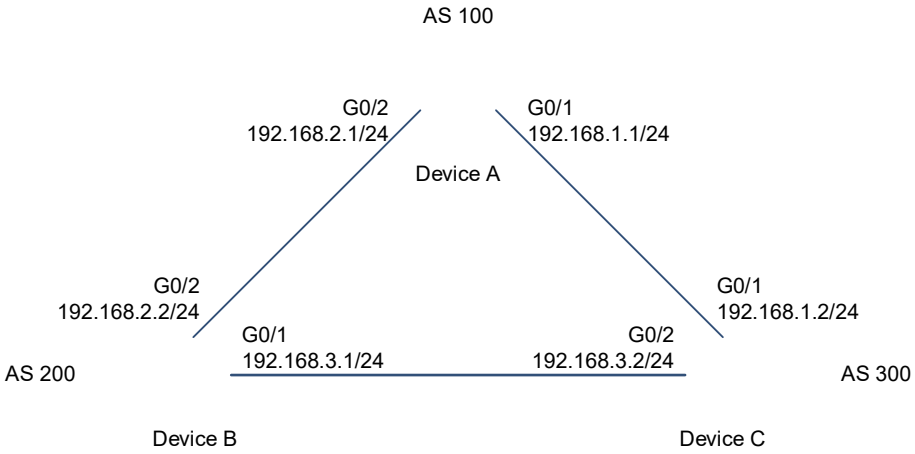
1.11.6 Configuring EBGP FRR

1. Requirements

The FRR function needs to be configured to back up and forward routing information and reduce the delay in service switching.

2. Topology

Figure 1-1 Topology of EBGP FRR



3. Notes

- Enable BGP on all devices (omitted).
- Configure addresses for directly connected interfaces on devices A, B, and C, and configure them to establish EBGP neighbor relationships (omitted).
- Configure a BFD session for the EBGP neighbor relationship between devices B and C.
- Configure FRR on device C.

4. Procedure

(1) Configure interface addresses and interface BFD parameters on devices B and C.

Configure device B.

```
Device B# configure terminal
Device B(config)# interface GigabitEthernet 0/1
Device B(config-if-GigabitEthernet 0/1)# ip address 192.168.3.1 255.255.255.0
Device B(config-if-GigabitEthernet 0/1)# bfd interval 200 min_rx 200
multiplier 5
Device B(config-if-GigabitEthernet 0/1)# exit
Device B(config)# interface GigabitEthernet 0/2
Device B(config-if-GigabitEthernet 0/2)# ip address 192.168.2.2 255.255.255.0
Device B(config-if-GigabitEthernet 0/2)# exit
```

Configure device C.

```
Device C# configure terminal
Device C(config)# interface GigabitEthernet 0/1
Device C(config-if-GigabitEthernet 0/1)# ip address 192.168.1.2 255.255.255.0
Device C(config-if-GigabitEthernet 0/1)# exit
Device C(config)# interface GigabitEthernet 0/2
```

```
Device C(config-if-GigabitEthernet 0/2)# ip address 192.168.3.2 255.255.255.0
Device C(config-if-GigabitEthernet 0/2)# bfd interval 200 min_rx 200
multiplier 5
Device C(config-if-GigabitEthernet 0/2)# exit
```

(2) Configure BFD for BGP and BGP FRR.

Configure device B.

```
Device B# configure terminal
Device B(config)# router bgp 200
Device B(config-router)# neighbor 192.168.3.2 remote-as 300
Device B(config-router)# neighbor 192.168.3.2 fall-over bfd
Device B(config-router)# neighbor 192.168.2.1 remote-as 100
Device B(config-router)# redistribute connect
```

Configure device C.

```
Device C# configure terminal
Device C(config)# router bgp 300
Device C(config-router)# neighbor 192.168.1.1 remote-as 100
Device C(config-router)# neighbor 192.168.3.1 remote-as 200
Device C(config-router)# neighbor 192.168.3.1 fall-over bfd
Device C(config-router)# address-family ipv4 unicast
Device C(config-router-af)# bgp fast-reroute
Device C(config-router-af)# redistribute connect
```

5. Verification

On device C, check BGP information.

```
Device C# show ip bgp summary
BGP router identifier 10.10.10.10, local AS number 300
BGP table version is 12
4 BGP AS-PATH entries
0 BGP Community entries
3 BGP Prefix entries (Maximum-prefix:4294967295)

Neighbor        V           AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down
State/PfxRcd
192.168.1.1     4          100      76      77       12    12    0 00:59:27
3
192.168.3.1     4          200      30      30       12    12    0 00:19:03
3

Total number of neighbors 2

Device C# show ip bgp
BGP table version is 12, local router ID is 10.10.10.10
Status codes: s suppressed, d damped, h history, * valid, > best, i -
internal,
              S Stale, b - backup entry
```

```
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric      LocPrf      Weight Path
*  192.168.1.0      192.168.3.1              0                        0 200 ?
*                   192.168.1.1              0                        0 100 ?
*>                  0.0.0.0                  0                    32768     ?
*> 192.168.2.0      192.168.3.1              0                        0 200 ?
*b                  192.168.1.1              0                        0 100 ?
*  192.168.3.0      192.168.3.1              0                        0 200 ?
*                   192.168.1.1              0                        0 100 200
?
*>                  0.0.0.0                  0                    32768     ?

Total number of prefixes 3
Device C# show ip bgp 192.168.2.0
BGP routing table entry for 192.168.2.0/24
Paths: (2 available, best #1, table Default-IP-Routing-Table)
  Advertised to non peer-group peers:
  192.168.1.1
  200
    192.168.3.1 from 192.168.3.1 (3.3.3.3)
      Origin incomplete, metric 0, localpref 100, valid, external, best
      Last update: Tue Oct  5 00:26:52 1971

  100
    192.168.1.1 from 192.168.1.1 (44.44.44.44)
      Origin incomplete, metric 0, localpref 100, valid, external, backup
      Last update: Mon Oct  4 23:46:28 1971
Device C# show ip route

Codes:  C - connected, S - static, R - RIP, B - BGP
        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
        i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
        ia - IS-IS inter area, * - candidate default

Gateway of last resort is no set
C    192.168.1.0/24 is directly connected, GigabitEthernet 1/9
C    192.168.1.2/32 is local host.
B    192.168.2.0/24 [20/0] via 192.168.3.1, 00:21:39
C    192.168.3.0/24 is directly connected, GigabitEthernet 1/11
C    192.168.3.2/32 is local host.
```

6. Configuration Files

- Device B configuration file

```
!
interface GigabitEthernet 0/1
 no switchport
 ip address 192.168.3.1 255.255.255.0
 bfd interval 200 min_rx 200 multiplier 5
!
interface GigabitEthernet 0/2
 no switchport
 ip address 192.168.2.2 255.255.255.0
!
router bgp 200
 bgp log-neighbor-changes
 bgp graceful-restart restart-time 120
 bgp graceful-restart stalepath-time 360
 bgp graceful-restart
 neighbor 192.168.2.1 remote-as 100
 neighbor 192.168.3.2 remote-as 300
 neighbor 192.168.3.2 fall-over bfd
 address-family ipv4
  redistribute connected
  neighbor 192.168.2.1 activate
  neighbor 192.168.3.2 activate
  exit-address-family
!
```

- Device C configuration file

```
!
interface GigabitEthernet 0/1
 no switchport
 ip address 192.168.1.2 255.255.255.0
 bfd interval 200 min_rx 200 multiplier 5
!
interface GigabitEthernet 0/2
 no switchport
 ip address 192.168.3.2 255.255.255.0
 bfd interval 200 min_rx 200 multiplier 5
!
router bgp 300
 bgp log-neighbor-changes
 bgp graceful-restart restart-time 120
 bgp graceful-restart stalepath-time 360
 bgp graceful-restart
 neighbor 192.168.1.1 remote-as 100
 neighbor 192.168.3.1 remote-as 200
 neighbor 192.168.3.1 fall-over bfd
 address-family ipv4
  bgp fast-reroute
```

```
  redistribute connected
  neighbor 192.168.1.1 activate
  neighbor 192.168.3.1 activate
  exit-address-family
!
```

1.11.7 Configuring BGP to Rapidly Withdraw Specified Routes

1. Requirements

Specified routing information needs to be withdrawn.

2. Topology

Figure 1-1 Topology of Fast Withdrawal of Specified Routes

3. Notes

- Enable BGP on all devices (omitted).
- Configure devices A and B to establish a BGP neighbor relationship (omitted).
- Configure an ACL on device A.
- Configure fast withdrawal of specified routes on device A.

4. Procedure

(1) Configure an ACL to permit the address 1.1.1.1.

Configure device A.

```
Device A# configure terminal
Device A(config)# ip access-list standard 1
Device A(config-std-nacl)# permit 1.1.1.1 0.0.0.0
Device A(config-std-nacl)# exit
```

(2) Configure fast withdrawal of the specified route 1.1.1.1.

Configure device A.

```
Device A# configure terminal
Device A(config)# router bgp 65530
Device A(config-router)# neighbor 192.168.1.2 remote-as 65531
Device A(config-router)# bgp fast-withdraw access-list 1
```

5. Verification

On device A, check BGP routing information.

```
Device A# show run router bgp
router bgp 65530
```

```
bgp log-neighbor-changes
bgp graceful-restart restart-time 120
bgp graceful-restart stalepath-time 360
bgp graceful-restart
neighbor 192.168.1.2 remote-as 65531

address-family ipv4
 bgp fast-withdraw access-list 1
 neighbor 192.168.1.2 activate
exit-address-family
```

6. Configuration Files

- Device A configuration file

```
!
ip access-list standard 1
 10 permit host 1.1.1.1
!
router bgp 65530
 bgp log-neighbor-changes
 bgp graceful-restart restart-time 120
 bgp graceful-restart stalepath-time 360
 bgp graceful-restart
 neighbor 192.168.1.2 remote-as 65531
 address-family ipv4
  bgp fast-withdraw access-list 1
  neighbor 192.168.1.2 activate
  exit-address-family
!
```

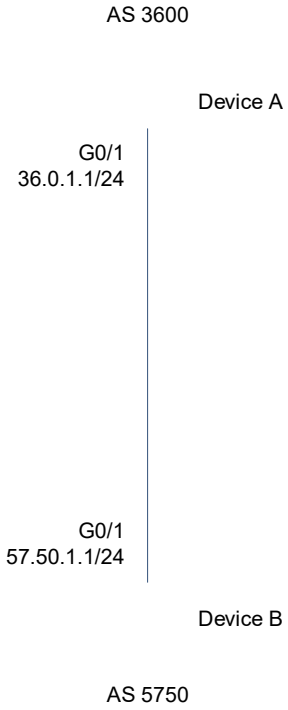
1.11.8 Configuring the BGP Local AS

1. Requirements

When the real AS of the local BGP changes, a new virtual AS 23 can be configured between peers so that the BGP configurations on peers do not need to be configured.

2. Topology

Figure 1-1 Topology of BGP Local AS



3. Notes

- Establish an EBGP neighbor relationship between devices A and B and set the local AS number of the EBGP neighbor to 23.
- Establish an EBGP neighbor relationship with devices A on B.

4. Procedure

Configure device A.

```
Device A# configure terminal
Device A(config)# router bgp 3600
Device A(config-router)# neighbor 57.50.1.1 remote-as 5750
Device A(config-router)# neighbor 57.50.1.1 update-source loopback 0
Device A(config-router)# neighbor 57.50.1.1 ebgp-multihop 255
Device A(config-router)# neighbor 57.50.1.1 local-as 23 no-prepend replace-as
dual-as
```

Configure device B.

```
Device B# configure terminal
Device B(config)# router bgp 5750
Device B(config-router)# neighbor 36.0.1.1 remote-as 23
Device B(config-router)# neighbor 36.0.1.1 update-source loopback 0
Device B(config-router)# neighbor 36.0.1.1 ebgp-multihop 255
```

5. Verification

On device A, check the BGP neighbor status.

```
Device A# show ip bgp neighbors 57.50.1.1
BGP neighbor is 57.50.1.1, remote AS 5750, local AS 23(using Peer's Local AS,
no-prepend, replace-as, dual-as), external link
  BGP version 4, remote router ID 0.0.0.0
  BGP state = Idle
  Last read, hold time is 180, keepalive interval is 60 seconds
  Received 0 messages, 0 notifications, 0 in queue
    open message:0 update message:0 keepalive message:0
    refresh message:0 dynamic cap:0 notifications:0
  Sent 0 messages, 0 notifications, 0 in queue
```

6. Configuration Files

- Device A configuration file

```
!
router bgp 3600
 bgp log-neighbor-changes
 bgp graceful-restart restart-time 120
 bgp graceful-restart stalepath-time 360
 bgp graceful-restart
 neighbor 57.50.1.1 remote-as 5750
 neighbor 57.50.1.1 local-as 23 no-prepend replace-as dual-as
 neighbor 57.50.1.1 ebgp-multihop 255
 neighbor 57.50.1.1 update-source Loopback 0
 address-family ipv4
  neighbor 57.50.1.1 activate
  exit-address-family
!
```

- Device B configuration file

```
!
router bgp 5750
 bgp log-neighbor-changes
 bgp graceful-restart restart-time 120
 bgp graceful-restart stalepath-time 360
 bgp graceful-restart
 neighbor 36.0.1.1 remote-as 23
 neighbor 36.0.1.1 ebgp-multihop 255
 neighbor 36.0.1.1 update-source Loopback 0
 address-family ipv4
  neighbor 36.0.1.1 activate
  exit-address-family
!
```

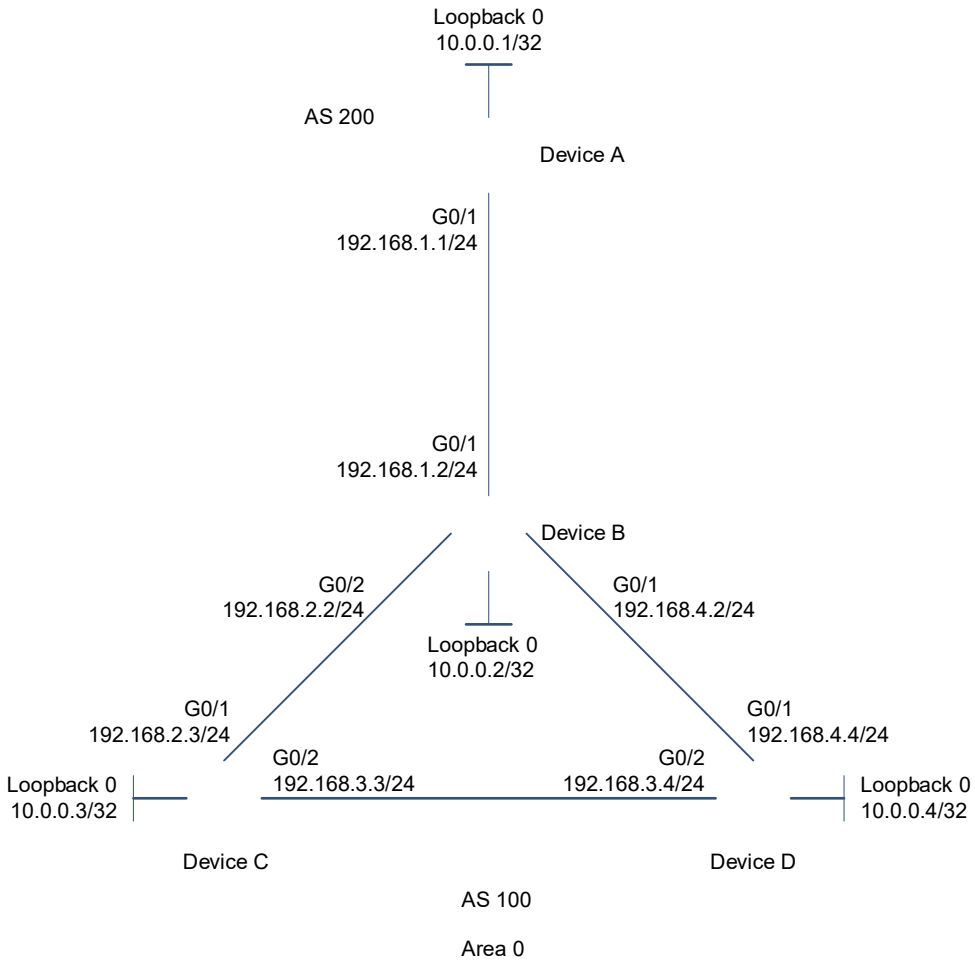
1.11.9 Configuring BGP GR

1. Requirements

Data forwarding should not be interrupted during the protocol restart.

2. Topology

Figure 1-1 Topology of BGP GR



3. Notes

- Enable BGP on all devices and set AS numbers for the devices as shown in Figure 1-1 (omitted).
- Configure a loopback interface on each of devices B, C, and D and configure the devices to establish IBGP neighbor relationships through the loopback interfaces (omitted).
- Configure device As and B to establish an EBGP neighbor relationship through directly connected interfaces (omitted).
- On devices A, B, C, and D, and enable the BGP GR function.

4. Procedure

Configure device A.

```
Device A# configure terminal
Device A(config)# router ospf 1
Device A(config-router)# graceful-restart
Device A(config-router)# exit
Device A(config)# router bgp 200
Device A(config-router)# bgp graceful-restart
```

Configure device B.

```
Device B# configure terminal
Device B(config)# router ospf 1
Device B(config-router)# graceful-restart
Device B(config-router)# exit
Device B(config)# router bgp 100
Device B(config-router)# bgp graceful-restart
```

Configure device C.

```
Device C# configure terminal
Device C(config)# router ospf 1
Device C(config-router)# graceful-restart
Device C(config-router)# exit
Device C(config)# router bgp 100
Device C(config-router)# bgp graceful-restart
```

Configure device D.

```
Device D# configure terminal
Device D(config)# router ospf 1
Device D(config-router)# graceful-restart
Device D(config-router)# exit
Device D(config)# router bgp 100
Device D(config-router)# bgp graceful-restart
```

5. Verification

On device B, run the **show ip ospf** command to display the OSPF routing status.

```
Device B# show ip ospf
Routing Process "ospf 1" with ID 10.0.0.2
Process uptime is 4 minutes
Process bound to VRF default
Conforms to RFC2328, and RFC1583Compatibility flag isenabled
Supports only single TOS(TOS0) routes
Supports opaque LSA
This router is an ASBR (injecting external routing information)
SPF schedule delay 5 secs, Hold time between two SPFs 10 secs
LsaGroupPacing: 240 secs
Number of incomming current DD exchange neighbors 0/5
Number of outgoing current DD exchange neighbors 0/5
Number of external LSA 4. Checksum 0x0278E0
Number of opaque AS LSA 0. Checksum 0x000000
Number of non-default external LSA 4
```

```
External LSA database is unlimited.
Number of LSA originated 6
Number of LSA received 2
Log Neighbor Adjency Changes : Enabled
Graceful-restart enabled
Graceful-restart helper support enabled
Number of areas attached to this router: 1
Area 0 (BACKBONE)
```

On device B, run the **show ip bgp neighbors** command to display the BGP neighbor status.

```
Device B# show ip bgp neighbors
BGP neighbor is 192.168.195.183, remote AS 200, local AS 100, external link
Using BFD to detect fast fallover - BFD session state up
  BGP version 4, remote router ID 10.0.0.1
  BGP state = Established, up for 00:06:37
  Last read 00:06:37, hold time is 180, keepalive interval is 60 seconds
  Neighbor capabilities:
    Route refresh: advertised and received (old and new)
Address family IPv4 Unicast: advertised and received
Graceful restart: advertised and received
  Remote Restart timer is 120 seconds
  Address families preserved by peer:
    None
```

6. Configuration Files

- Device A configuration file

```
router bgp 200
 bgp log-neighbor-changes
 bgp graceful-restart restart-time 120
 bgp graceful-restart stalepath-time 360
 bgp graceful-restart
 address-family ipv4
  exit-address-family
!
router ospf 1
 graceful-restart
!
```

- Device B configuration file

```
router bgp 100
 bgp log-neighbor-changes
 bgp graceful-restart restart-time 120
 bgp graceful-restart stalepath-time 360
 bgp graceful-restart
 address-family ipv4
  exit-address-family
!
```

```
router ospf 1
 graceful-restart
!
```

- Device C configuration file

```
router bgp 100
 bgp log-neighbor-changes
 bgp graceful-restart restart-time 120
 bgp graceful-restart stalepath-time 360
 bgp graceful-restart
 address-family ipv4
  exit-address-family
!
router ospf 1
 graceful-restart
!
```

- Device D configuration file

```
router bgp 100
 bgp log-neighbor-changes
 bgp graceful-restart restart-time 120
 bgp graceful-restart stalepath-time 360
 bgp graceful-restart
 address-family ipv4
  exit-address-family
!
router ospf 1
 graceful-restart
!
```

7. Common Errors

- GR is not enabled for IGP.
- GR is not enabled for a BGP neighbor.

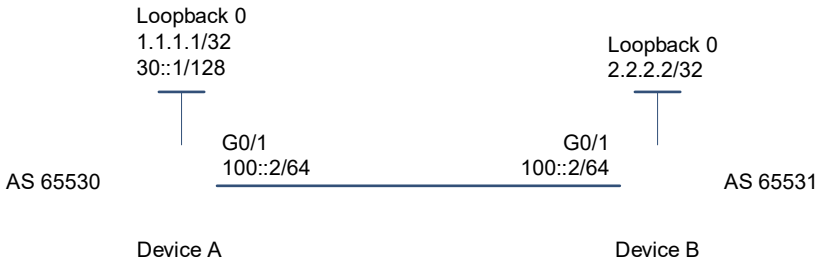
1.11.10 Configuring IPv6 Route Exchange Across ASs

1. Requirements

IPv6 routing information needs to be exchanged across different ASs.

2. Topology

Figure 1-1 Topology of IPv6 Route Exchange Across ASs



3. Notes

- Enable BGP on all devices and set AS numbers for the devices as shown in Figure 1-1 (omitted).
- Configure BGP neighbors, disable the IPv4 address family capability for the neighbors, and activate the IPv6 address family capability.
- Configure BGP to advertise IPv6 routes.

4. Procedure

(1) Configure BGP and activate the IPv6 address family.

Configure device A.

```
Device A# configure terminal
Device A(config)# router bgp 65530
Device A(config-router)# neighbor 100::1 remote-as 65531
Device A(config-router)# address-family ipv4
Device A(config-router-af)# no neighbor 100::1 activate
Device A(config-router-af)# exit-address-family
Device A(config-router)# address-family ipv6
Device A(config-router-af)# neighbor 100::1 activate
```

Configure device B.

```
Device B# configure terminal
Device B(config)# router bgp 65531
Device B(config-router)# neighbor 100::2 remote-as 65530
Device B(config-router)# address-family ipv4
Device B(config-router-af)# no neighbor 100::2 activate
Device B(config-router-af)# exit-address-family
Device B(config-router)# address-family ipv6
Device B(config-router-af)# neighbor 100::2 activate
```

(2) Configure the advertisement of IPv6 network information.

Configure device A.

```
Device A# configure terminal
Device A(config)# router bgp 65530
Device A(config-router)# address-family ipv6
```

```
Device A(config-router-af)# network 30::1/128
```

5. Verification

On device A, check the BGP neighbor status.

```
Device A# show bgp ipv6 unicast summary
BGP router identifier 1.1.1.1, local AS number 65530
BGP table version is 1
1 BGP AS-PATH entries
0 BGP Community entries
1 BGP Prefix entries (Maximum-prefix:4294967295)

Neighbor        V           AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down
State/PfxRcd
100::1          4        65531       4       6        1    0    0 00:01:49
0

Total number of neighbors 1
```

On device B, check the BGP neighbor status.

```
Device B# 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, b - backup entry
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric      LocPrf      Weight Path
*> 30::1/128        100::2                   0                       0 65530 i

Total number of prefixes 1
```

6. Configuration Files

- Device A configuration file

```
!
router bgp 65530
 bgp log-neighbor-changes
 bgp graceful-restart restart-time 120
 bgp graceful-restart stalepath-time 360
 bgp graceful-restart
 neighbor 100::1 remote-as 65531
 address-family ipv4
  no neighbor 100::1 activate
  exit-address-family
 address-family ipv6
  network 30::1/128
  neighbor 100::1 activate
  exit-address-family
```

```
!
```

- Device B configuration file

```
!
router bgp 65531
 bgp log-neighbor-changes
 bgp graceful-restart restart-time 120
 bgp graceful-restart stalepath-time 360
 bgp graceful-restart
 neighbor 100::2 remote-as 65530
 address-family ipv4
  no neighbor 100::2 activate
  exit-address-family
 address-family ipv6
  neighbor 100::2 activate
  exit-address-family
!
```

7. Common Errors

- The IPv6 address family capability is not activated for BGP neighbors.
- In non-6PE scenarios, IPv4 addresses are used to establish a neighbor relationship to exchange IPv6 routes.

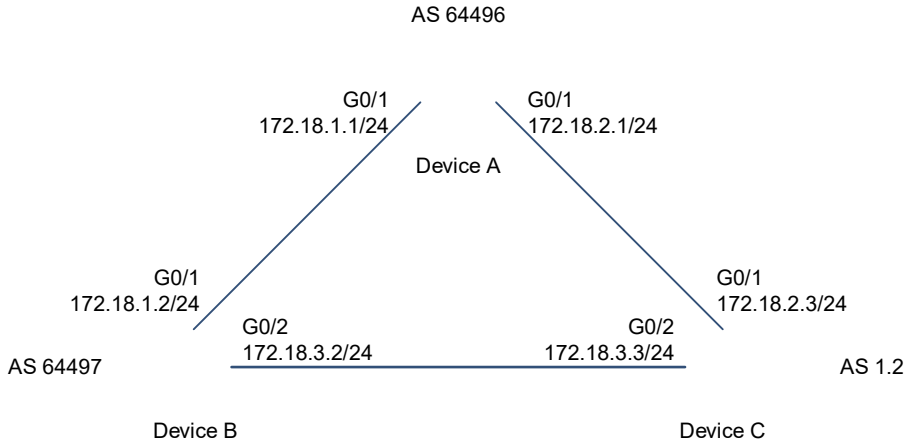
1.11.11 Configuring Compatibility Between BGP Devices Supporting 4-Byte AS Numbers and Those Supporting 2-Byte AS Numbers

1. Requirements

- Routers supporting 2-byte AS numbers and those supporting 4-byte AS numbers (using 2-byte AS numbers) can establish BGP connections.
- Routers supporting 2-byte AS numbers and those supporting 4-byte AS numbers (using 4-byte AS numbers) can establish BGP connections.
- Two routers supporting 4-byte AS numbers can establish a BGP connection, with one using a 2-byte AS number and the other using a 4-byte AS number.

2. Topology

Figure 1-1 Topology of Compatibility Between BGP Devices Supporting 4-Byte AS Numbers and Those Supporting 2-Byte AS Numbers



3. Notes

- Enable BGP on all devices and set AS numbers for the devices as shown in Figure 1-1 (omitted).
- Configure BGP neighbors.

4. Procedure

Configure device A.

```
Device A# configure terminal
Device A(config)# router bgp 64496
Device A(config-router)# neighbor 172.18.1.2 remote-as 64497
Device A(config-router)# neighbor 172.18.2.3 remote-as 23456
Device A(config-router)# end
```

Configure device B.

```
Device B# configure terminal
Device B(config)# router bgp 64497
Device B(config-router)# neighbor 172.18.1.1 remote-as 64496
Device B(config-router)# neighbor 172.18.3.3 remote-as 1.2
Device B(config-router)# bgp asnotation dot
Device B(config-router)# end
```

Configure device C.

```
Device C# configure terminal
Device C(config)# router bgp 1.2
Device C(config-router)# neighbor 172.18.2.1 remote-as 64496
Device C(config-router)# neighbor 172.18.3.2 remote-as 64497
Device C(config-router)# end
```

5. Verification

On device A, check the BGP neighbor status.

```
Device A# show ip bgp summary

BGP router identifier 172.18.1.1, local AS number 64496
BGP table version is 1, main routing table version 1

Neighbor     V   AS      MsgRcvd   MsgSent   TblVer   InQ   OutQ   Up/Down
Statd
172.18.1.2   4   64497   7         7         1        0     0      00:03:04
0
172.18.2.3   4   23456   4         4         1        0     0      00:00:15
0
```

On device B, check the BGP neighbor status.

```
Device B# show ip bgp summary

BGP router identifier 172.18.3.2, local AS number 64497
BGP table version is 1, main routing table version 1

Neighbor     V   AS      MsgRcvd   MsgSent   TblVer   InQ   OutQ   Up/Down
Statd
172.18.1.1   4   64496   7         7         1         0    0      00:00:04
0
172.18.3.2   4   1.2     4         4         1         0    0      00:00:16
0
```

6. Configuration Files

- Device A configuration file

```
!
router bgp 64496
 bgp log-neighbor-changes
 bgp graceful-restart restart-time 120
 bgp graceful-restart stalepath-time 360
 bgp graceful-restart
 neighbor 172.18.1.2 remote-as 64497
 neighbor 172.18.2.3 remote-as 23456
 address-family ipv4
  neighbor 172.18.1.2 activate
  neighbor 172.18.2.3 activate
  exit-address-family
!
```

- Device B configuration file

```
!
router bgp 64497
```

```
 bgp asnotation dot
 bgp log-neighbor-changes
 bgp graceful-restart restart-time 120
 bgp graceful-restart stalepath-time 360
 bgp graceful-restart
 neighbor 172.18.1.1 remote-as 64496
 neighbor 172.18.3.3 remote-as 1.2
 address-family ipv4
  neighbor 172.18.1.1 activate
  neighbor 172.18.3.3 activate
  exit-address-family
!
```

- Device C configuration file

```
!
router bgp 65538
 bgp log-neighbor-changes
 bgp graceful-restart restart-time 120
 bgp graceful-restart stalepath-time 360
 bgp graceful-restart
 neighbor 172.18.2.1 remote-as 64496
 neighbor 172.18.3.2 remote-as 64497
 address-family ipv4
  neighbor 172.18.2.1 activate
  neighbor 172.18.3.2 activate
  exit-address-family
!
```

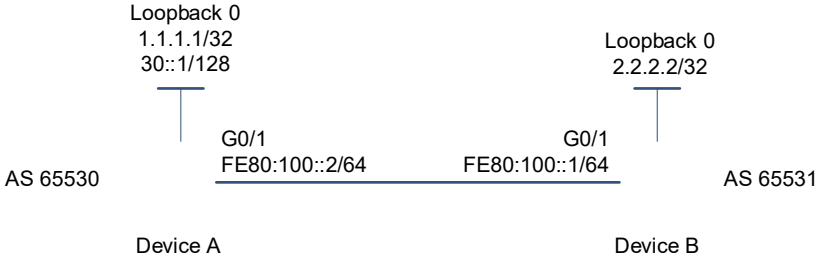
1.11.12 Configuring an IPv6 Local Link Address

1. Requirements

An IPv6 local link address needs to be configured as a neighbor update address.

2. Topology

Figure 1-1 Topology of IPv6 Local Link Address



3. Notes

- Enable BGP on all devices, configure IPv6 local link addresses for interfaces, and set the AS numbers for

the devices as shown in Figure 1-1 (omitted).

- Configure BGP neighbors, disable the IPv4 address family capability for the neighbors, and activate the IPv6 address family capability.
- Configure BGP to advertise IPv6 routes.

4. Procedure

(1) Enable the IPv6 address family capability and disable the IPv4 address family capability.

Configure device A.

```
Device A# configure terminal
Device A(config)# router bgp 65530
Device A(config-router)# neighbor fe80:100::1%Gi0/1 remote-as 65531
Device A(config-router)# address-family ipv4
Device A(config-router-af)# no neighbor fe80:100::1%Gi0/1 activate
Device A(config-router-af)# exit-address-family
Device A(config-router)# address-family ipv6
Device A(config-router-af)# neighbor fe80:100::1%Gi0/1 activate
Device A(config-router-af)# network 30::1/128
```

Configure device B.

```
Device B# configure terminal
Device B(config)# router bgp 65531
Device B(config-router)# neighbor fe80:100::2%Gi0/1 remote-as 65530
Device B(config-router)# address-family ipv4
Device B(config-router-af)# no neighbor fe80:100::2%Gi0/1 activate
Device B(config-router-af)# exit-address-family
Device B(config-router)# address-family ipv6
Device B(config-router-af)# neighbor fe80:100::2%Gi0/1 activate
```

(2) Configure BGP to advertise IPv6 addresses.

Configure device A.

```
Device A# configure terminal
Device A(config)# router bgp 65530
Device A(config-router)# address-family ipv6
Device A(config-router-af)# network 30::1/128
```

5. Verification

On device A, check the BGP neighbor status.

```
Device A# show bgp ipv6 unicast summary
BGP router identifier 1.1.1.1, local AS number 65530
BGP table version is 1
1 BGP AS-PATH entries
0 BGP Community entries
1 BGP Prefix entries (Maximum-prefix:4294967295)

Neighbor              V          AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down
State/PfxRcd
```

```
fe80:100::1%Gi0/1     4       65531       4       6        1    0    0
00:01:49        0

Total number of neighbors 1
```

On device B, check the BGP neighbor status.

```
Device B# 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, b - backup entry
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric      LocPrf      Weight Path
*> 30::1/128        fe80:100::2          0                        0 65530 i

Total number of prefixes 1
```

6. Configuration Files

- Device A configuration file

```
!
router bgp 65530
 bgp log-neighbor-changes
 bgp graceful-restart restart-time 120
 bgp graceful-restart stalepath-time 360
 bgp graceful-restart
 neighbor fe80:100::1%Gi0/1 remote-as 65531
 address-family ipv4
  no neighbor fe80:100::1%Gi0/1 activate
  exit-address-family
 address-family ipv6
  network 30::1/128
  neighbor fe80:100::1%Gi0/1 activate
  exit-address-family
!
```

- Device B configuration file

```
!
router bgp 65531
 bgp log-neighbor-changes
 bgp graceful-restart restart-time 120
 bgp graceful-restart stalepath-time 360
 bgp graceful-restart
 neighbor fe80:100::2%Gi0/1 remote-as 65530
 address-family ipv4
  no neighbor fe80:100::2%Gi0/1 activate
  exit-address-family
```

```
 address-family ipv6
  neighbor fe80:100::2%Gi0/1 activate
  exit-address-family
!
```

7. Common Errors

- An IPv6 local link address is used to establish a neighbor relationship during neighbor configuration, but the interface, on which the IPv6 local link address is configured, is not configured as the update source during local configuration.
- Only one end uses an IPv6 local link address for establishing a neighbor relationship.

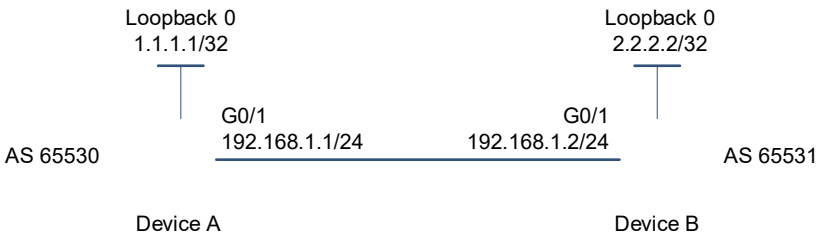
1.11.13 Configuring BGP NSR

1. Requirements

NSR can be configured to ensure that data switching and forwarding are not interrupted.

2. Topology

Figure 1-1 Topology of BGP NSR



3. Notes

- Enable BGP on all devices and set AS numbers for the devices as shown in Figure 1-1 (omitted).
- Configure devices A and B to establish an EBGP neighbor relationship through directly connected interfaces.
- Enable the BGP NSR function on device A.

4. Procedure

Configure device A.

```
Device A# configure terminal
Device A(config)# router bgp 65530
Device A(config-router)# neighbor 192.168.1.2 remote-as 65530
Device A(config-router)# neighbor 192.168.1.2 ha-mode nsr
```

5. Verification

On device A, check the BGP neighbor status.

```
Device A# show ip bgp neighbors
BGP neighbor is 192.168.1.2, remote AS 65530, local AS 65531, external link
Using BFD to detect fast fallover - BFD session state up
  BGP version 4, remote router ID 2.2.2.2
```

```
  BGP state = Established, up for 00:06:37
  Last read 00:06:37, hold time is 180, keepalive interval is 60 seconds
  Neighbor capabilities:
    Route refresh: advertised and received (old and new)
Address family IPv4 Unicast: advertised and received
Graceful restart: advertised and received
  Remote Restart timer is 120 seconds
  Address families preserved by peer:
    None
```

On device B, check the BGP neighbor status.

```
Device B# show ip bgp neighbors
BGP neighbor is 192.168.1.1, remote AS 65530, local AS 65531, external link
Using BFD to detect fast fallover - BFD session state up
  BGP version 4, remote router ID 1.1.1.1
  BGP state = Established, up for 00:06:37
  Last read 00:06:37, hold time is 180, keepalive interval is 60 seconds
  Neighbor capabilities:
    Route refresh: advertised and received (old and new)
Address family IPv4 Unicast: advertised and received
Graceful restart: advertised and received
  Remote Restart timer is 120 seconds
  Address families preserved by peer:
    None
```

6. Configuration Files

- Device A configuration file

```
!
router bgp 65530
 bgp log-neighbor-changes
 bgp graceful-restart restart-time 120
 bgp graceful-restart stalepath-time 360
 bgp graceful-restart
 bgp nsr
 neighbor 192.168.1.2 remote-as 65530
 address-family ipv4
  neighbor 192.168.1.2 activate
  exit-address-family
!
```

7. Common Errors

- IGP is not configured.
- The NSR function is not enabled for BGP neighbors.

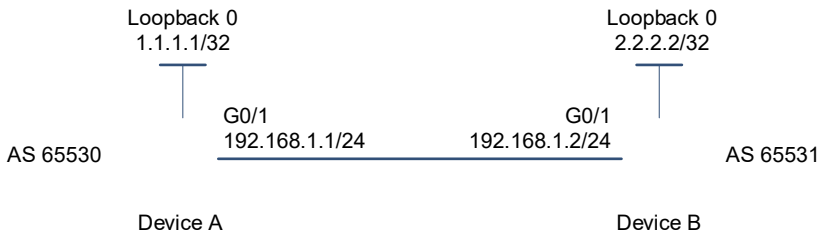
1.11.14 Configuring BGP Routes to Be Recursive Only to Host Routes

1. Requirements

Non-direct BGP routes need to be recursive to host routes.

2. Topology

Figure 1-1 Topology of Configuring BGP Routes to Be Recursive Only to Host Routes



3. Notes

- Enable BGP on all devices and set the AS numbers as shown in the figure above (omitted).
- Configure devices A and B to establish an EBGP neighbor relationship through loopback interfaces (omitted).
- On devices A and B, enable the function of making BGP routes recursive only to host routes.

4. Procedure

Configure device A.

```
Device A# configure terminal
Device A(config)# router bgp 65530
Device A(config-router)# bgp recursion host
```

Configure device B.

```
Device B# configure terminal
Device B(config)# router bgp 65531
Device B(config-router)# bgp recursion host
```

5. Configuration Files

- Device A configuration file

```
!
router bgp 65530
 bgp log-neighbor-changes
 bgp graceful-restart restart-time 120
 bgp graceful-restart stalepath-time 360
 bgp graceful-restart
 address-family ipv4
  bgp recursion host
  exit-address-family
!
```

- Device B configuration file

```
!
router bgp 65531
 bgp log-neighbor-changes
```

```
bgp graceful-restart restart-time 120
bgp graceful-restart stalepath-time 360
bgp graceful-restart
address-family ipv4
 bgp recursion host
 exit-address-family
!
```

6. Common Errors

If an IBGP or multi-hop EBGP neighbor relationship is established not using loopback addresses, after the function is enabled, BGP cannot to find host routes matching the next-hop addresses. As a result, the next hops of BGP routes are invalid and routes cannot be preferentially selected.

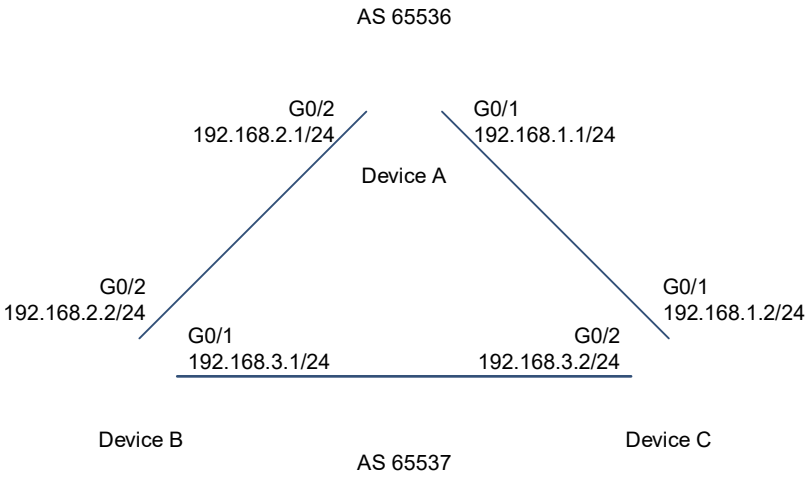
1.11.15 Configuring Outbound Loop Detection for a BGP Neighbor

1. Requirements

Outbound loop detection needs to be configured for a BGP neighbor to prevent the receiving of routing information carrying the AS number of the neighbor in the **AS-PATH** attribute.

2. Topology

Figure 1-1 Topology of Outbound Loop Detection for a BGP Neighbor



3. Notes

- Enable BGP on all devices and set AS numbers for the devices as shown in Figure 1-1 (omitted).
- Configure devices A and C and devices B and C to establish EBGP neighbor relationships (omitted).
- On device C, enable outbound loop detection for its neighbors: devices A and B.

4. Procedure

Configure device A.

```
Device A# configure terminal
Device A(config)# router bgp 65536
```

```
Device A(config-router)# neighbor 192.168.1.2 remote-as 65537
Device A(config-router)# neighbor 192.168.1.2 as-loop-check out
Device A(config-router)# neighbor 192.168.2.2 remote-as 65537
Device A(config-router)# neighbor 192.168.2.2 as-loop-check out
```

Configure device B.

```
Device B# configure terminal
Device B(config)# router bgp 65537
Device B(config-router)# neighbor 192.168.2.1 remote-as 65536
Device B(config-router)# neighbor 192.168.3.2 remote-as 65537
```

Configure device C.

```
Device C# configure terminal
Device C(config)# router bgp 65537
Device C(config-router)# neighbor 192.168.1.1 remote-as 65536
Device C(config-router)# neighbor 192.168.3.1 remote-as 65537
```

5. Verification

On device A, check the BGP neighbor status.

```
Device A# show ip bgp neighbors 192.168.2.2
BGP neighbor is 192.168.2.2, remote AS 65537, local AS 65536, external link
Using as path loop detection in announcing route
  BGP version 4, remote router ID 192.168.2.2
  BGP state = Established, up for 00:06:37
  Last read 00:06:37, hold time is 180, keepalive interval is 60 seconds
  Neighbor capabilities:
    Route refresh: advertised and received (old and new)
Address family IPv4 Unicast: advertised and received
Graceful restart: advertised and received
  Remote Restart timer is 120 seconds
  Address families preserved by peer:
    None
......
```

6. Configuration Files

- Device A configuration file

```
!
router bgp 65536
 bgp log-neighbor-changes
 bgp graceful-restart restart-time 120
 bgp graceful-restart stalepath-time 360
 bgp graceful-restart
 neighbor 192.168.1.2 remote-as 65537
 neighbor 192.168.2.2 remote-as 65537
 address-family ipv4
  neighbor 192.168.1.2 activate
  neighbor 192.168.1.2 as-loop-check out
```

```
  neighbor 192.168.2.2 activate
  neighbor 192.168.2.2 as-loop-check out
  exit-address-family
!
```

- Device B configuration file

```
!
router bgp 65537
 bgp log-neighbor-changes
 bgp graceful-restart restart-time 120
 bgp graceful-restart stalepath-time 360
 bgp graceful-restart
 neighbor 192.168.2.1 remote-as 65536
 neighbor 192.168.3.2 remote-as 65537
 address-family ipv4
  neighbor 192.168.2.1 activate
  neighbor 192.168.3.2 activate
  exit-address-family
!
```

- Device C configuration file

```
!
router bgp 65537
 bgp log-neighbor-changes
 bgp graceful-restart restart-time 120
 bgp graceful-restart stalepath-time 360
 bgp graceful-restart
 neighbor 192.168.1.1 remote-as 65536
 neighbor 192.168.3.1 remote-as 65537
 address-family ipv4
  neighbor 192.168.1.1 activate
  neighbor 192.168.3.1 activate
  exit-address-family
!
```

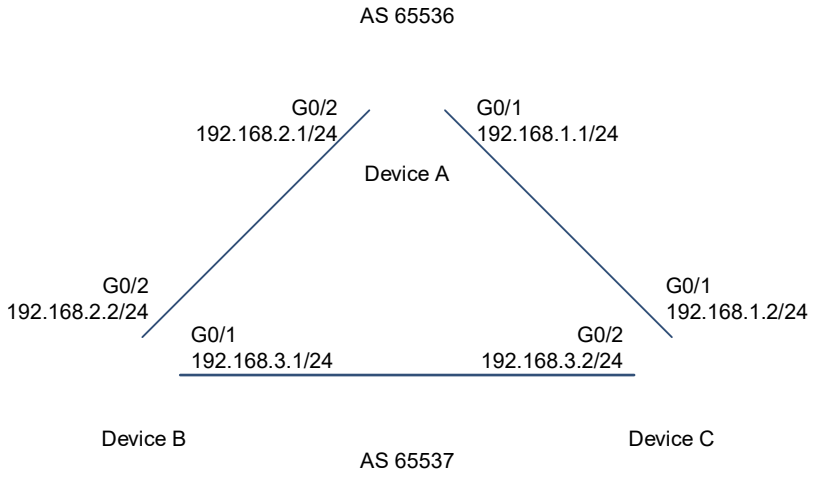
1.11.16 Shutting Down BGP Connections Gracefully

1. Requirements

Graceful shutdown of BGP connections needs to be configured on a device. Then, after a certain period of time (automatically calculated based on the number of advertised routes or specified by *delay time*), the device disconnects BGP connections from its neighbors to reduce the impact on the network.

2. Topology

Figure 1-1 Topology of Graceful Shutdown of BGP Connections



3. Notes

- Enable BGP on all devices and set AS numbers for the devices as shown in Figure 1-1 (omitted).
- Configure devices A and C and devices B and C to establish EBGP neighbor relationships (omitted).
- On device A, configure the graceful shutdown of all connections in the BGP instance.

4. Procedure

Configure device A.

```
Device A# configure terminal
Device A(config)# router bgp 65536
Device A(config-router)# neighbor 192.168.1.2 remote-as 65537
Device A(config-router)# neighbor 192.168.1.2 as-loop-check out
Device A(config-router)# neighbor 192.168.2.2 remote-as 65537
Device A(config-router)# neighbor 192.168.2.2 as-loop-check out
```

Configure device B.

```
Device B# configure terminal
Device B(config)# router bgp 65537
Device B(config-router)# neighbor 192.168.2.1 remote-as 65536
Device B(config-router)# neighbor 192.168.3.2 remote-as 65537
```

Configure device C.

```
Device C# configure terminal
Device C(config)# router bgp 65537
Device C(config-router)# neighbor 192.168.1.1 remote-as 65536
Device C(config-router)# neighbor 192.168.3.1 remote-as 65537
Device C(config-router)# bgp shutdown graceful
```

5. Verification

On device C, check the BGP neighbor status.

```
Device C(config)#show ip bgp neighbor 192.168.1.1
For address family: IPv4 Unicast
BGP neighbor is 192.168.1.1, remote AS 65536, local AS 65537, external link
 Administratively graceful shut down
  BGP version 4, remote router ID 0.0.0.0
  BGP state = Idle
  Last read         , hold time is 30, keepalive interval is 1 seconds
  Configured hold time is 30, keepalive interval is 1 seconds
  Received 0 messages, 0 notifications, 0 in queue
    open message:0 update message:0 keepalive message:0
    refresh message:0 dynamic cap:0 notifications:0
  Sent 0 messages, 0 notifications, 0 in queue
    open message:0 update message:0 keepalive message:0
    refresh message:0 dynamic cap:0 notifications:0
  Route refresh request: received 0, sent 0
  Minimum time between advertisement runs is 30 seconds
  Update source is Loopback 0

 For address family: IPv4 Unicast
  BGP table version 1, neighbor version 0
  Index 1, Offset 0, Mask 0x2
  0 accepted prefixes
  0 announced prefixes

 Connections established 0; dropped 0
  BGP neighbor may be up to 255 hops away.
Sock_fd: -1
Last Reset:         , due to BGP Notification received
Notification Error Message: (Cease/Other Configuration Change.)
```

6. Configuration Files

- Device A configuration file

```
!
router bgp 65536
 bgp log-neighbor-changes
 bgp graceful-restart restart-time 120
 bgp graceful-restart stalepath-time 360
 bgp graceful-restart
 neighbor 192.168.1.2 remote-as 65537
 neighbor 192.168.2.2 remote-as 65537
 address-family ipv4
  neighbor 192.168.1.2 activate
  neighbor 192.168.1.2 as-loop-check out
```

```
  neighbor 192.168.2.2 activate
  neighbor 192.168.2.2 as-loop-check out
  exit-address-family
!
```

- Device B configuration file

```
!
router bgp 65537
 bgp log-neighbor-changes
 bgp graceful-restart restart-time 120
 bgp graceful-restart stalepath-time 360
 bgp graceful-restart
 neighbor 192.168.2.1 remote-as 65536
 neighbor 192.168.3.2 remote-as 65537
 address-family ipv4
  neighbor 192.168.2.1 activate
  neighbor 192.168.3.2 activate
  exit-address-family
!
```

- Device C configuration file

```
!
router bgp 65537
 bgp shutdown graceful
 bgp log-neighbor-changes
 bgp graceful-restart restart-time 120
 bgp graceful-restart stalepath-time 360
 bgp graceful-restart
 neighbor 192.168.1.1 remote-as 65536
 neighbor 192.168.3.1 remote-as 65537
 address-family ipv4
  neighbor 192.168.1.1 activate
  neighbor 192.168.3.1 activate
  exit-address-family
!
```

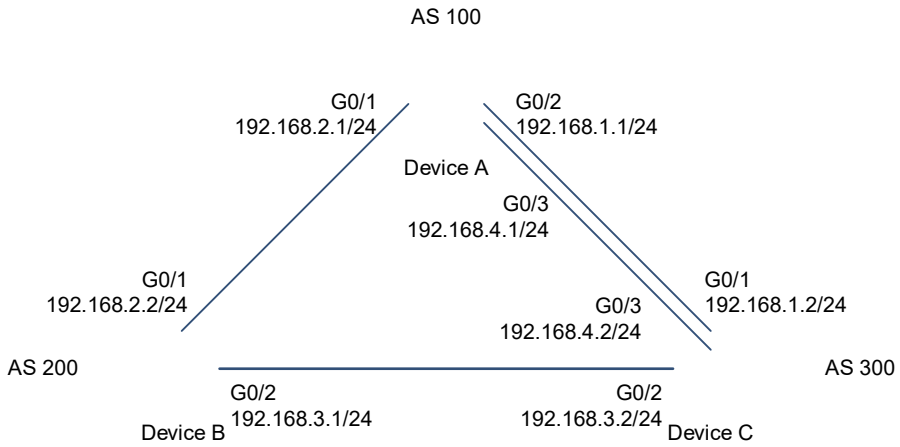
1.11.17 Configuring BGP Multi-Path Bypass Protection

1. Requirements

Multi-path bypass protection needs to be configured to enhance routing reliability.

2. Topology

Figure 1-1 Topology of BGP Multi-Path Bypass Protection



3. Notes

- Enable BGP on all devices (omitted).
- Configure addresses for directly connected interfaces on devices A, B, and C, and configure them to establish EBGP neighbor relationships (omitted).
- Configure a BFD session for the EBGP neighbor relationship between devices A and C (omitted).
- On device C, configure BGP multi-path bypass protection and ECMP.

4. Procedure

Configure device A.

```
Device A# configure terminal
Device A(config)# router bgp 100
Device A(config-router)# neighbor 192.168.1.2 remote-as 300
Device A(config-router)# neighbor 192.168.4.2 remote-as 300
Device A(config-router)# neighbor 192.168.2.2 remote-as 200
Device A(config-router)# maximum-paths ebgp 2
Device A(config-router)# redistribute connect
```

Configure device B.

```
Device B# configure terminal
Device B(config)# router bgp 200
Device B(config-router)# neighbor 192.168.3.2 remote-as 300
Device B(config-router)# neighbor 192.168.3.2 fall-over bfd
Device B(config-router)# neighbor 192.168.2.1 remote-as 100
Device B(config-router)# redistribute connect
```

Configure device C.

```
Device C# configure terminal
```

```
Device C(config)# router bgp 300
Device C(config-router)# neighbor 192.168.1.1 remote-as 100
Device C(config-router)# neighbor 192.168.4.1 remote-as 100
Device C(config-router)# neighbor 192.168.3.1 remote-as 200
Device C(config-router)# neighbor 192.168.1.1 fall-over bfd
Device C(config-router)# neighbor 192.168.4.1 fall-over bfd
Device C(config-router)# maximum-paths ebgp 2
Device C(config-router)# address-family ipv4 unicast
Device C(config-router-af)# bgp install standby-path
Device C(config-router-af)# redistribute connect
```

5. Verification

On device C, check BGP information.

```
Device C# show ip bgp summary
BGP router identifier 192.168.4.2, local AS number 300
BGP table version is 12
4 BGP AS-PATH entries
0 BGP Community entries
4 BGP Prefix entries (Maximum-prefix:4294967295)

Neighbor        V           AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down
State/PfxRcd
192.168.1.1     4          100      76      77       12    12    0 00:59:27
3
192.168.3.1     4          200      30      30       12    12    0 00:19:03
3
192.168.4.1     4          100      76      77       12    12    0 00:59:01
3

Total number of neighbors 3

Device C# show ip bgp
BGP table version is 12, local router ID is 10.10.10.10
Status codes: s suppressed, d damped, h history, * valid, > best, i -
internal,
              S Stale, b - backup entry, m - multipath
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric      LocPrf      Weight Path
*  192.168.1.0      192.168.3.1              0                       0 200 ?
*                   192.168.4.1              0                       0 100 ?
*                   192.168.1.1              0                       0 100 ?
*>                  0.0.0.0                  0                   32768     ?
*> 192.168.2.0      192.168.4.1              0                       0 200 ?
*m                  192.168.1.1              0                       0 200 ?
*b                  192.168.3.1              0                       0 100 ?
```

```
*  192.168.3.0      192.168.3.1              0                        0 200 ?
*                   192.168.4.1              0                        0 100 200
?
*                   192.168.1.1              0                        0 100 200
?
*>                  0.0.0.0                  0                    32768     ?
*  192.168.4.0      192.168.3.1              0                        0 200 ?
*                   192.168.4.1              0                        0 100 ?
*                   192.168.1.1              0                        0 100 ?
*>                  0.0.0.0                  0                    32768     ?

Total number of prefixes 4
Device C# show ip bgp 192.168.2.0
BGP routing table entry for 192.168.2.0/24
Paths: (3 available, best #3, table Default-IP-Routing-Table)
  Advertised to non peer-group peers:
  192.168.1.1
  200
    192.168.3.1 from 192.168.3.1 (3.3.3.3)
      Origin incomplete, metric 0, localpref 100, valid, external, backup
      Last update: Tue Oct  5 00:26:52 1971

  100
    192.168.1.1 from 192.168.1.1 (44.44.44.44)
      Origin incomplete, metric 0, localpref 100, valid, external, multipath
      Last update: Mon Oct  4 23:46:28 1971

  100
    192.168.4.1 from 192.168.4.1 (44.44.44.44)
      Origin incomplete, metric 0, localpref 100, valid, external, multipath,
best
      Last update: Mon Oct  4 23:46:28 1971
Device C# show ip route

Codes:  C - connected, S - static, R - RIP, B - BGP
        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
        i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
        ia - IS-IS inter area, * - candidate default

Gateway of last resort is no set
C    192.168.1.0/24 is directly connected, GigabitEthernet 0/1
C    192.168.1.2/32 is local host.
B    192.168.2.0/24 [20/0] via 192.168.1.1, 00:21:39
                    [20/0] via 192.168.4.1, 00:21:39
```

```
C    192.168.3.0/24 is directly connected, GigabitEthernet 0/2
C    192.168.3.2/32 is local host.
C    192.168.4.0/24 is directly connected, GigabitEthernet 0/3
C    192.168.4.2/32 is local host.
```

6. Configuration Files

- Device A configuration file

```
!
router bgp 100
 bgp log-neighbor-changes
 bgp graceful-restart restart-time 120
 bgp graceful-restart stalepath-time 360
 bgp graceful-restart
 neighbor 192.168.1.2 remote-as 300
 neighbor 192.168.2.2 remote-as 200
 neighbor 192.168.4.2 remote-as 300
 address-family ipv4
  maximum-paths ebgp 2
  redistribute connected
  neighbor 192.168.1.2 activate
  neighbor 192.168.2.2 activate
  neighbor 192.168.4.2 activate
  exit-address-family
!
```

- Device B configuration file

```
!
router bgp 200
 bgp log-neighbor-changes
 bgp graceful-restart restart-time 120
 bgp graceful-restart stalepath-time 360
 bgp graceful-restart
 neighbor 192.168.2.1 remote-as 100
 neighbor 192.168.3.2 remote-as 300
 neighbor 192.168.3.2 fall-over bfd
 address-family ipv4
  redistribute connected
  neighbor 192.168.2.1 activate
  neighbor 192.168.3.2 activate
  exit-address-family
!
```

- Device C configuration file

```
!
router bgp 300
 bgp log-neighbor-changes
 bgp graceful-restart restart-time 120
```

```
bgp graceful-restart stalepath-time 360
bgp graceful-restart
neighbor 192.168.1.1 remote-as 100
neighbor 192.168.1.1 fall-over bfd
neighbor 192.168.3.1 remote-as 200
neighbor 192.168.4.1 remote-as 100
neighbor 192.168.4.1 fall-over bfd
address-family ipv4
 maximum-paths ebgp 2
 bgp install standby-path
 redistribute connected
 neighbor 192.168.1.1 activate
 neighbor 192.168.3.1 activate
 neighbor 192.168.4.1 activate
 exit-address-family
!
```

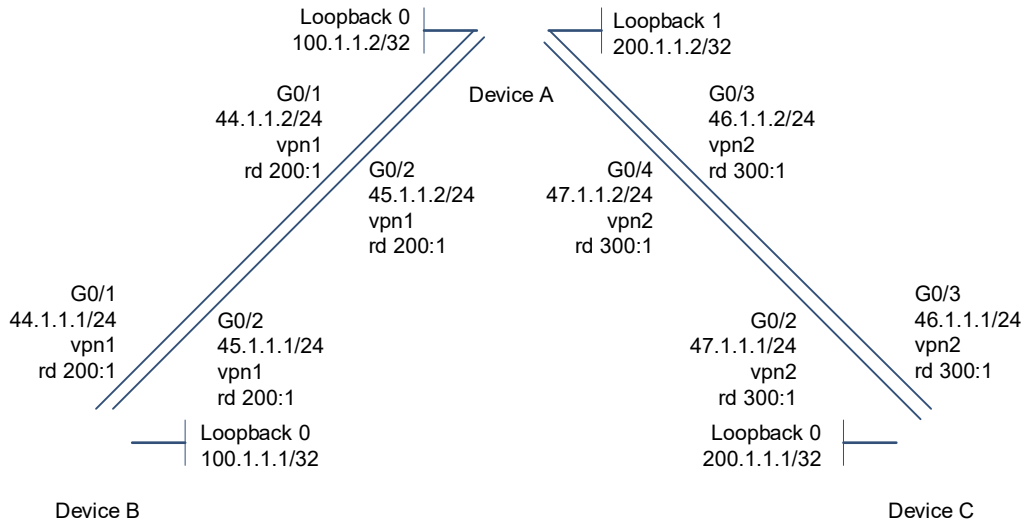
1.11.18 Configuring Multi-Path Route Import Between VRF Instances

1. Requirements

Device A is connected with two networks through the interfaces of two VRF instances (vpn1 and vpn2) so that cross-VRF access is implemented through device A.

2. Topology

Figure 1-1 Topology of Multi-Path Route Import Between VRF Instances



3. Notes

- Configure VRF instances and VRF static routes.
- Configure a VRF address family.
- Import VRF static routes to BGP.
- Enable the function of importing multi-path static routes to BGP.

- Configure BGP ECMP.
- Configure the import of all path routes between VRF instances.

4. Procedure

(1) Configure VRF instances and VRF static routes.

Configure device A.

```
Device A# configure terminal
Device A(config)# ip vrf vpn1
Device A(config-vrf)# rd 200:1
Device A(config-vrf)# route-target both 100:100
Device A(config-vrf)# exit
Device A(config)# ip vrf vpn2
Device A(config-vrf)# rd 300:1
Device A(config-vrf)# route-target both 100:100
Device A(config-vrf)# exit
Device A(config)# interface GigabitEthernet 0/1
Device A(config-if-GigabitEthernet 0/1)# ip vrf forwarding vpn1
Device A(config-if-GigabitEthernet 0/1)# ip address 44.1.1.2 255.255.255.0
Device A(config-if-GigabitEthernet 0/1)# exit
Device A(config)# interface GigabitEthernet 0/2
Device A(config-if-GigabitEthernet 0/2)# ip vrf forwarding vpn1
Device A(config-if-GigabitEthernet 0/2)# ip address 45.1.1.2 255.255.255.0
Device A(config-if-GigabitEthernet 0/2)# exit
Device A(config)# interface GigabitEthernet 0/3
Device A(config-if-GigabitEthernet 0/3)# ip vrf forwarding vpn2
Device A(config-if-GigabitEthernet 0/3)# ip address 46.1.1.2 255.255.255.0
Device A(config-if-GigabitEthernet 0/3)# exit
Device A(config)# interface GigabitEthernet 0/4
Device A(config-if-GigabitEthernet 0/4)# ip vrf forwarding vpn2
Device A(config-if-GigabitEthernet 0/4)# ip address 47.1.1.2 255.255.255.0
Device A(config-if-GigabitEthernet 0/4)# exit
Device A(config)# ip route vrf vpn1 100.1.1.1 255.255.255.255 44.1.1.1
Device A(config)# ip route vrf vpn1 100.1.1.1 255.255.255.255 45.1.1.1
Device A(config)# ip route vrf vpn2 200.1.1.1 255.255.255.255 46.1.1.1
Device A(config)# ip route vrf vpn2 200.1.1.1 255.255.255.255 47.1.1.1
```

Configure device B.

```
Device B# configure terminal
Device B(config)# ip vrf vpn1
Device B(config-vrf)# rd 200:1
Device B(config-vrf)# route-target both 100:100
Device B(config-vrf)# exit
Device B(config)# interface GigabitEthernet 0/1
Device B(config-if-GigabitEthernet 0/1)# ip vrf forwarding vpn1
Device B(config-if-GigabitEthernet 0/1)# ip address 44.1.1.1 255.255.255.0
Device B(config-if-GigabitEthernet 0/1)# exit
```

```
Device B(config)# interface GigabitEthernet 0/2
Device B(config-if-GigabitEthernet 0/2)# ip vrf forwarding vpn1
Device B(config-if-GigabitEthernet 0/2)# ip address 45.1.1.1 255.255.255.0
Device B(config-if-GigabitEthernet 0/2)# exit
Device B(config)# ip route vrf vpn1 100.1.1.2 255.255.255.255 44.1.1.2
Device B(config)# ip route vrf vpn1 100.1.1.2 255.255.255.255 45.1.1.2
```

Configure device C.

```
Device C# configure terminal
Device C(config)# ip vrf vpn2
Device C(config-vrf)# rd 300:1
Device C(config-vrf)# route-target both 100:100
Device C(config-vrf)# exit
Device C(config)# interface GigabitEthernet 0/1
Device C(config-if-GigabitEthernet 0/1)# ip vrf forwarding vpn2
Device C(config-if-GigabitEthernet 0/1)# ip address 46.1.1.1 255.255.255.0
Device C(config-if-GigabitEthernet 0/1)# exit
Device C(config)# interface GigabitEthernet 0/2
Device C(config-if-GigabitEthernet 0/2)# ip vrf forwarding vpn2
Device C(config-if-GigabitEthernet 0/2)# ip address 47.1.1.1 255.255.255.0
Device C(config-if-GigabitEthernet 0/2)# exit
Device C(config)# ip route vrf vpn2 200.1.1.2 255.255.255.255 46.1.1.2
Device C(config)# ip route vrf vpn2 200.1.1.2 255.255.255.255 47.1.1.2
```

(2) Configure the function of importing multi-path static routes to BGP to implement inter-VRF route import.

Configure device A.

```
Device A# configure terminal
Device A(config)# router bgp 100
Device A(config-router)# address-family ipv4 vrf vpn1
Device A(config-router-af)# redistribute static
Device A(config-router-af)# maximum-paths ebgp 32
Device A(config-router-af)# bgp sourced-paths static all
Device A(config-router-af)# import path selection all
Device A(config-router-af)# exit-address-family
Device A(config-router)# address-family ipv4 vrf vpn2
Device A(config-router-af)# redistribute static
Device A(config-router-af)# maximum-paths ebgp 32
Device A(config-router-af)# bgp sourced-paths static all
Device A(config-router-af)# import path selection all
Device A(config-router-af)# exit-address-family
```

5. Verification

On device A, check VRF routes.

```
Device A# show ip route vrf vpn1
Routing Table: vpn1

Codes:  C - Connected, L - Local, S - Static
```

```
        R - RIP, O - OSPF, B - BGP, I - IS-IS, V - Overflow route
        N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
        E1 - OSPF external type 1, E2 - OSPF external type 2
        SU - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
        IA - Inter area, EV - BGP EVPN, A - Arp to host
        * - candidate default

Gateway of last resort is no set
C     44.1.1.0/24 is directly connected, GigabitEthernet 0/1
C     44.1.1.2/32 is local host.
C     45.1.1.0/24 is directly connected, GigabitEthernet 0/2
C     45.1.1.2/32 is local host.
S     100.1.1.1/32 [1/0] via 44.1.1.1
                   [1/0] via 45.1.1.1
B     200.1.1.1/32 [20/0] via 47.1.1.1, 02:32:01
                   [20/0] via 46.1.1.1, 02:32:01

Device A# show ip route vrf vpn2
Routing Table: vpn2

Codes:  C - Connected, L - Local, S - Static
        R - RIP, O - OSPF, B - BGP, I - IS-IS, V - Overflow route
        N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
        E1 - OSPF external type 1, E2 - OSPF external type 2
        SU - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
        IA - Inter area, EV - BGP EVPN, A - Arp to host
        * - candidate default

Gateway of last resort is no set
C     46.1.1.0/24 is directly connected, GigabitEthernet 0/3
C     46.1.1.2/32 is local host.
C     47.1.1.0/24 is directly connected, GigabitEthernet 0/4
C     47.1.1.2/32 is local host.
B     100.1.1.1/32 [20/0] via 45.1.1.1, 03:27:07
                   [20/0] via 44.1.1.1, 03:27:07
S     200.1.1.1/32 [1/0] via 46.1.1.1
                   [1/0] via 47.1.1.1
```

6. Configuration Files

- Device A configuration file

```
!
ip vrf vpn1
 rd 200:1
 route-target both 100:100
!
ip vrf vpn2
```

```
 rd 300:1
 route-target both 100:100
!
interface GigabitEthernet 0/1
 no switchport
 ip vrf forwarding vpn1
 ip address 44.1.1.2 255.255.255.0
!
interface GigabitEthernet 0/2
 no switchport
 ip vrf forwarding vpn1
 ip address 45.1.1.2 255.255.255.0
!
interface GigabitEthernet 0/3
 no switchport
 ip vrf forwarding vpn2
 ip address 46.1.1.2 255.255.255.0
!
interface GigabitEthernet 0/4
 no switchport
 ip vrf forwarding vpn2
 ip address 47.1.1.2 255.255.255.0
!
router bgp 100
 bgp log-neighbor-changes
 bgp graceful-restart restart-time 120
 bgp graceful-restart stalepath-time 360
 bgp graceful-restart
 address-family ipv4
  exit-address-family
  !
 address-family ipv4 vrf vpn1
  maximum-paths ebgp 32
  bgp sourced-paths static all
  import path selection all
  redistribute static
  exit-address-family
  !
 address-family ipv4 vrf vpn2
  maximum-paths ebgp 32
  bgp sourced-paths static all
  import path selection all
  redistribute static
  exit-address-family
!
ip route vrf vpn1 100.1.1.1 255.255.255.255 44.1.1.1
```

```
ip route vrf vpn1 100.1.1.1 255.255.255.255 45.1.1.1
ip route vrf vpn2 200.1.1.1 255.255.255.255 46.1.1.1
ip route vrf vpn2 200.1.1.1 255.255.255.255 47.1.1.1
!
```

- Device B configuration file

```
!
ip vrf vpn1
 rd 200:1
 route-target both 100:100
!
interface GigabitEthernet 0/1
 no switchport
 ip vrf forwarding vpn1
 ip address 44.1.1.1 255.255.255.0
!
interface GigabitEthernet 0/2
 no switchport
 ip vrf forwarding vpn1
 ip address 45.1.1.1 255.255.255.0
!
ip route vrf vpn1 100.1.1.2 255.255.255.255 44.1.1.2
ip route vrf vpn1 100.1.1.2 255.255.255.255 45.1.1.2
!
```

- Device C configuration file

```
!
ip vrf vpn2
 rd 300:1
 route-target both 100:100
!
interface GigabitEthernet 0/1
 no switchport
 ip vrf forwarding vpn2
 ip address 46.1.1.1 255.255.255.0
!
interface GigabitEthernet 0/2
 no switchport
 ip vrf forwarding vpn2
 ip address 47.1.1.1 255.255.255.0
!
ip route vrf vpn2 200.1.1.2 255.255.255.255 46.1.1.2
ip route vrf vpn2 200.1.1.2 255.255.255.255 47.1.1.2
!
```

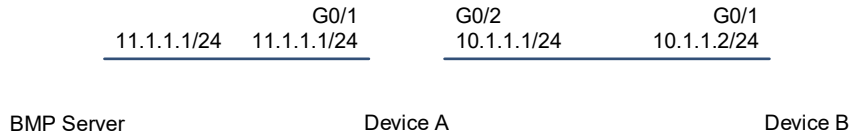
1.11.19 Configuring BMP Monitoring

1. Requirements

The BGP connections established between devices A and B need to be monitored by the BMP server.

2. Topology

Figure 1-1 Topology of BMP Monitoring



3. Notes

- Configure BGP basic features (omitted).
- Configure information relevant to a BMP server instance (address, port number, and type of monitored routes).
- Configure the BMP server to monitor BGP peers.

4. Procedure

(1) Configure information related to a BMP server instance.

Configure device A.

```
Device A# configure
Device A(config)# bmp server 1
Device A(config-bmpsrvr)# address 11.1.1.2 port 12345
Device A(config-bmpsrvr)# update-source gigabitEthernet 0/1
Device A(config-bmpsrvr)# adj-rib-in post-policy
```

(2) Configure the BMP server to monitor BGP peers.

Configure device A.

```
Device A(config)# router bgp 100
Device A(config-router)# neighbor 10.1.1.2 remote-as 100
Device A(config-router)# neighbor 10.1.1.2 bmp-active server 1
```

5. Verification

On device A, check BGP information.

```
Device A# show bgp all summary
For address family: IPv4 Unicast
BGP router identifier 11.1.1.1, local AS number 100
BGP table version is 1
1 BGP AS-PATH entries
0 BGP Community entries
25 BGP Prefix entries (Maximum-prefix:262144)
```

```
Neighbor        V           AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down
State/PfxRcd
10.1.1.2        4          100       0       0        0    0    0 05:36:32
23

Device A# show bgp bmp neighbor

Neighbor        CfgSvr#           ActSvr#
10.1.1.2        1                 1

Device A# show bgp bmp summary
ID   Host                Port     State        Time        NBRs
1    11.1.1.2            12345    Established  05:36:12    1
```

6. Configuration Files

- Device A configuration file

```
!
router bgp 100
 bgp log-neighbor-changes
 bgp graceful-restart restart-time 120
 bgp graceful-restart stalepath-time 360
 bgp graceful-restart
 neighbor 10.1.1.2 remote-as 100
 neighbor 10.1.1.2 bmp-active server 1
 address-family ipv4
  neighbor 10.1.1.2 activate
  exit-address-family
!
bmp server 1
 address 11.1.1.2 port 12345
 update-source GigabitEthernet 0/1
 adj-rib-in post-policy
!
```

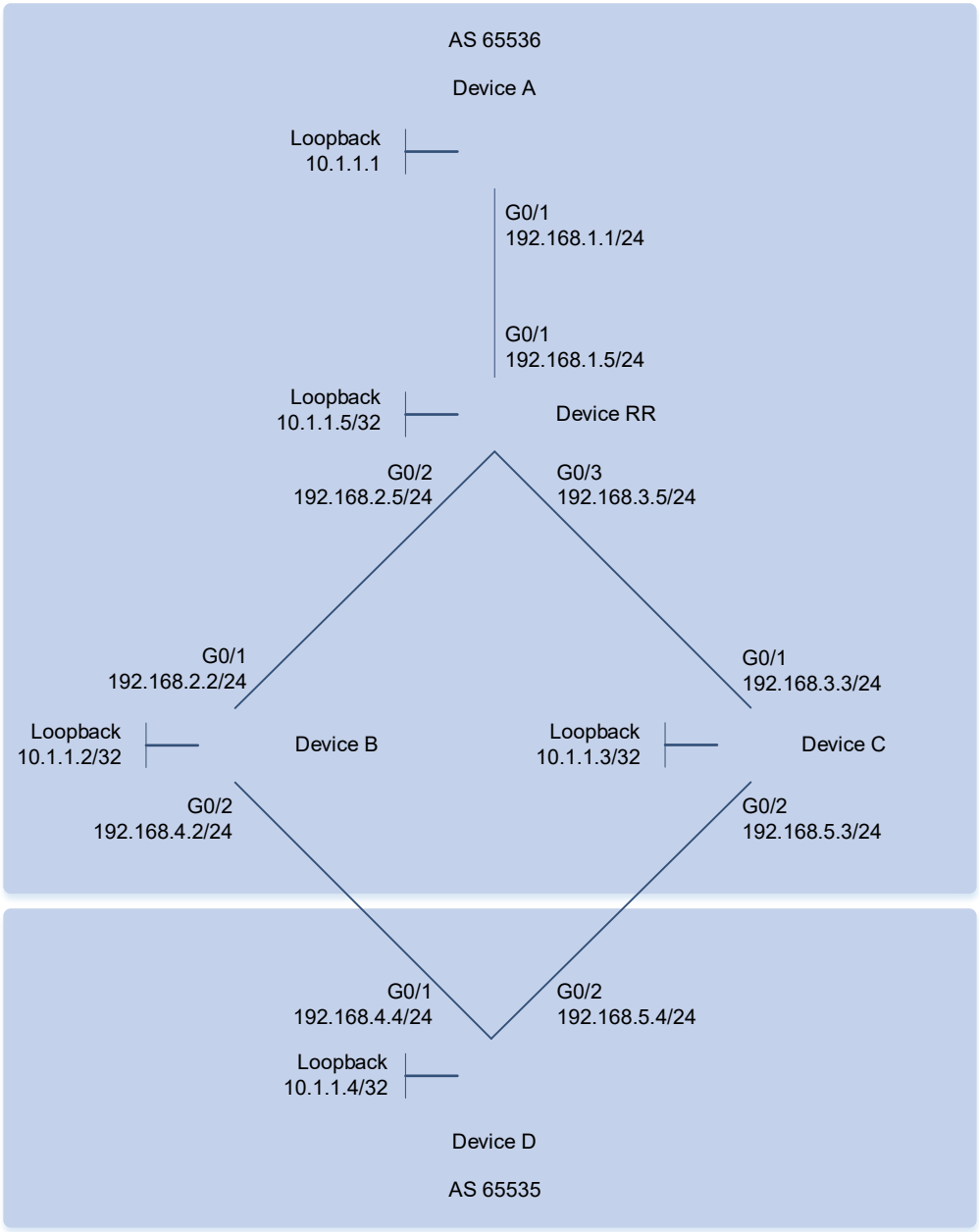
1.11.20 Configuring BGP ADD-PATH

1. Requirements

When the device is allowed to advertise multiple routes with the same prefix to a neighbor, you can define ADD-PATH to enable multiple links to implement load sharing or backup, thereby increasing the network link reliability.

2. Topology

Figure 1-1 Topology of the BGP ADD-PATH



3. Notes

- As shown in Figure 1-1, configure interconnection addresses for devices A, B, C, D, and RR, configure devices A, B, C, and RR to establish IBGP neighbor relationships, configure device RR as a BGP route reflector, and configure devices B and D and devices C and D to establish EBGP neighbor relationships.
- Configure a loopback address on device D and advertise it to BGP.
- Configure the ADD-PATH advertisement capability on device RR and the ADD-PATH receiving capability on device A.

4. Procedure

(1) Configure BGP basic features and interface IP addresses on devices A, B, C, and D.

Configure device A.

```
Device A# configure terminal
Device A(config)# interface GigabitEthernet 0/1
Device A(config-if-GigabitEthernet 0/1)# ip address 192.168.1.1 255.255.255.0
Device A(config-if-GigabitEthernet 0/1)# exit
Device A(config)# router bgp 65536
Device A(config-router)# neighbor 192.168.1.5 remote-as 65536
Device A(config-router)# neighbor 192.168.1.5 additional-paths receive
```

Configure device B.

```
Device B# configure terminal
Device B(config)# interface GigabitEthernet 0/1
Device B(config-if-GigabitEthernet 0/1)# ip address 192.168.2.2 255.255.255.0
Device B(config-if-GigabitEthernet 0/1)# exit
Device B(config)# interface GigabitEthernet 0/2
Device B(config-if-GigabitEthernet 0/2)# ip address 192.168.4.2 255.255.255.0
Device B(config-if-GigabitEthernet 0/2)# exit
Device B(config)# router bgp 65536
Device B(config-router)# neighbor 192.168.2.5 remote-as 65536
Device B(config-router)# neighbor 192.168.4.4 remote-as 65535
```

Configure device C.

```
Device C# configure terminal
Device C(config)# interface GigabitEthernet 0/1
Device C(config-if-GigabitEthernet 0/1)# ip address 192.168.3.3 255.255.255.0
Device C(config-if-GigabitEthernet 0/1)# exit
Device C(config)# interface GigabitEthernet 0/2
Device C(config-if-GigabitEthernet 0/2)# ip address 192.168.5.3 255.255.255.0
Device C(config-if-GigabitEthernet 0/2)# exit
Device C(config)# router bgp 65536
Device C(config-router)# neighbor 192.168.3.5 remote-as 65536
Device C(config-router)# neighbor 192.168.5.4 remote-as 65535
```

Configure device D.

```
Device D# configure terminal
Device D(config)# interface loopback 0
Device D(config-if-Loopback 0)# ip address 10.1.1.4 255.255.255.255
Device D(config-if-Loopback 0)# exit
Device D(config)# interface GigabitEthernet 0/1
Device D(config-if-GigabitEthernet 0/1)# ip address 192.168.4.4 255.255.255.0
Device D(config-if-GigabitEthernet 0/1)# exit
Device D(config)# interface GigabitEthernet 0/1
Device D(config-if-GigabitEthernet 0/1)# ip address 192.168.5.4 255.255.255.0
Device D(config-if-GigabitEthernet 0/1)# exit
Device D(config)# router bgp 65535
```

```
Device D(config-router)# neighbor 192.168.4.2 remote-as 65536
Device D(config-router)# neighbor 192.168.5.3 remote-as 65536
Device D(config-router)# network 10.1.1.4 mask 255.255.255.255
```

Configure device RR.

```
Device RR# configure terminal
Device RR(config)# interface loopback 0
Device RR(config-if-Loopback 0)# ip address 10.1.1.5 255.255.255.255
Device RR(config-if-Loopback 0)# exit
Device RR(config)# interface GigabitEthernet 0/1
Device RR(config-if-GigabitEthernet 0/1)# ip address 192.168.1.5 255.255.255.0
Device RR(config-if-GigabitEthernet 0/1)# exit
Device RR(config)# interface GigabitEthernet 0/2
Device RR(config-if-GigabitEthernet 0/2)# ip address 192.168.2.5 255.255.255.0
Device RR(config-if-GigabitEthernet 0/2)# exit
Device RR(config)# interface GigabitEthernet 0/3
Device RR(config-if-GigabitEthernet 0/3)# ip address 192.168.3.5 255.255.255.0
Device RR(config-if-GigabitEthernet 0/3)# exit
Device RR(config)#ip route 192.168.4.0 255.255.255.0 192.168.2.2
Device RR(config)#ip route 192.168.5.0 255.255.255.0 192.168.3.3
Device RR(config)# router bgp 65536
Device RR(config-router)# neighbor 192.168.1.1 remote-as 65536
Device RR(config-router)# neighbor 192.168.1.1 route-reflector-client
Device RR(config-router)# neighbor 192.168.2.2 remote-as 65536
Device RR(config-router)# neighbor 192.168.2.2 route-reflector-client
Device RR(config-router)# neighbor 192.168.3.3 remote-as 65536
```

(2) Configure the ADD-PATH rules for the route reflector RR.

Configure device RR.

```
Device RR# configure terminal
Device RR(config)# router bgp 65536
Device RR(config-router)# bgp additional-paths select best 2
Device RR(config-router)# neighbor 192.168.1.1 advertise additional-paths best
2
Device RR(config-router)# neighbor 192.168.1.1 additional-paths send
```

5. Verification

On device RR, check the BGP neighbor status.

```
Device RR# show ip bgp
BGP table version is 3, local router ID is 10.1.1.5
Status codes: s suppressed, d damped, h history, * valid, > best, i -
internal,
              S Stale, b - backup entry, a additional-path
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric      LocPrf      Weight Path
* ia10.1.1.4/32    192.168.4.4              0    100      0 65535 i
```

```
*>i                192.168.5.4              0    100      0 65535 i
Total number of prefixes 2
```

On device A, run the **show** command to display the BGP neighbor status.

```
Device A# show ip bgp
BGP table version is 2, local router ID is 192.168.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i -
internal,
              S Stale, b - backup entry, a additional-path
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric      LocPrf      Weight Path
* i 10.1.1.4/32    192.168.5.4              0    100      0 65535 i
* i                192.168.4.4              0    100      0 65535 i
Total number of prefixes 2
```

6. Configuration Files

- Device A configuration file

```
!
interface GigabitEthernet 0/1
 no switchport
 ip address 192.168.1.1 255.255.255.0
!
router bgp 65536
 bgp log-neighbor-changes
 bgp graceful-restart restart-time 120
 bgp graceful-restart stalepath-time 360
 bgp graceful-restart
 neighbor 192.168.1.5 remote-as 65536
 address-family ipv4
  neighbor 192.168.1.5 activate
  neighbor 192.168.1.5 additional-paths receive
  exit-address-family
!
```

- Device B configuration file

```
!
interface GigabitEthernet 0/1
 no switchport
 ip address 192.168.2.2 255.255.255.0
!
interface GigabitEthernet 0/2
 no switchport
 ip address 192.168.4.2 255.255.255.0
!
router bgp 65536
 bgp log-neighbor-changes
```

```
 bgp graceful-restart restart-time 120
 bgp graceful-restart stalepath-time 360
 bgp graceful-restart
 neighbor 192.168.2.5 remote-as 65536
 neighbor 192.168.4.4 remote-as 65535
 address-family ipv4
  neighbor 192.168.2.5 activate
  neighbor 192.168.4.4 activate
  exit-address-family
!
```

- Device C configuration file

```
!
interface GigabitEthernet 0/1
 no switchport
 ip address 192.168.3.3 255.255.255.0
!
interface GigabitEthernet 0/2
 no switchport
 ip address 192.168.5.3 255.255.255.0
!
router bgp 65536
 bgp log-neighbor-changes
 bgp graceful-restart restart-time 120
 bgp graceful-restart stalepath-time 360
 bgp graceful-restart
 neighbor 192.168.3.5 remote-as 65536
 neighbor 192.168.5.4 remote-as 65535
 address-family ipv4
  neighbor 192.168.3.5 activate
  neighbor 192.168.5.4 activate
  exit-address-family
!
```

- Device C configuration file

```
!
interface GigabitEthernet 0/1
 no switchport
 ip address 192.168.4.4 255.255.255.0
!
interface GigabitEthernet 0/2
 no switchport
 ip address 192.168.5.4 255.255.255.0
!
interface Loopback 0
 ip address 10.1.1.4 255.255.255.255
!
```

```
router bgp 65535
 bgp log-neighbor-changes
 bgp graceful-restart restart-time 120
 bgp graceful-restart stalepath-time 360
 bgp graceful-restart
 neighbor 192.168.4.2 remote-as 65536
 neighbor 192.168.5.3 remote-as 65536
 address-family ipv4
  network 10.1.1.4 mask 255.255.255.255
  neighbor 192.168.4.2 activate
  neighbor 192.168.5.3 activate
  exit-address-family
!
```

- Device RR configuration file

```
!
interface GigabitEthernet 0/1
 no switchport
 ip address 192.168.1.5 255.255.255.0
!
interface GigabitEthernet 0/2
 no switchport
 ip address 192.168.2.5 255.255.255.0
!
interface GigabitEthernet 0/3
 no switchport
 ip address 192.168.3.5 255.255.255.0
!
interface Loopback 0
 ip address 10.1.1.5 255.255.255.255
!
router bgp 65536
 bgp log-neighbor-changes
 bgp graceful-restart restart-time 120
 bgp graceful-restart stalepath-time 360
 bgp graceful-restart
 neighbor 192.168.1.1 remote-as 65536
 neighbor 192.168.2.2 remote-as 65536
 neighbor 192.168.3.3 remote-as 65536
 address-family ipv4
  bgp additional-paths select best 2
  neighbor 192.168.1.1 activate
  neighbor 192.168.1.1 route-reflector-client
  neighbor 192.168.1.1 additional-paths send
  neighbor 192.168.1.1 advertise additional-paths best 2
  neighbor 192.168.2.2 activate
  neighbor 192.168.2.2 route-reflector-client
```

```
  neighbor 192.168.3.3 activate
  exit-address-family
!
```