Configuring BGP

SYSTEM ADMINISTRATOR GUIDE

18/1543-CRA 119 1170/1-V1 Uen L

Contents

1 Overview

This document provides an overview of the Border Gateway Protocol (BGP) and describes the tasks and commands used to configure, monitor, troubleshoot, and administer BGP features through the SmartEdge router.

This document applies to both the Ericsson SmartEdge® and SM family routers. However, the software that applies to the SM family of systems is a subset of the SmartEdge OS; some of the functionality described in this document may not apply to SM family routers.

For information specific to the SM family chassis, including line cards, refer to the SM family chassis documentation.

For specific information about the differences between the SmartEdge and SM family routers, refer to the Technical Product Description *SM Family of Systems* (part number 5/221 02-CRA 119 1170/1) in the **`Product Overview`** folder of this Customer Product Information library.

BGP is an Exterior Gateway Protocol (EGP) based on distance-vector algorithms, and uses the Transmission Control Protocol (TCP) as its transport protocol. BGP is a protocol between two BGP nodes, or BGP speakers. First, the TCP connection is established and then the two BGP speakers exchange dynamic routing information over the connection. The exchange of messages is a BGP session between BGP peers.

The SmartEdge router supports multiple BGP features, including those specified in the following IETF drafts and RFCs:

- Base features:
  - Y. Rekhter, T. Li, RFC 4271, *Border Gateway Protocol 4 (BGP-4)*, January 2006
  - Y. Rekhter, T. Li, Internet Draft, *A Border Gateway Protocol 4 (BGP-4),* draft-ietf-idr-bgp4-12.txt, January 2001
- Route reflection:

  T. Bates, R. Chandra, E. Chen, RFC 2796, *BGP Route Reflection - An Alternative to Full Mesh IBGP*, April 2000
- Autonomous system confederations:

  P. Traina, D. McPherson, J. Scudder, RFC 3065, *Autonomous System Confederations for BGP*, February 2001
- Extended Communities attribute:

R. Chandra, P. Traina, T. Li, RFC 1997, *BGP Communities Attribute*, August 1996

- MD-5 authentication:

  A. Heffernan, RFC 2385, *Protection of BGP Sessions via the TCP MD5 Signature Option*, August 1998

- Route-flap damping:

  C. Villamizar, R. Chandra, R. Govindan, RFC 2439, *BGP Route Flap Damping*, November 1998

- Capabilities advertisement:

  R. Chandra, J. Scudder, RFC 2842, *Capabilities Advertisement with BGP-4*, May 2000

- Multiprotocol extensions:

  T. Bates, R. Chandra, D. Katz, Y. Rekhter, RFC 2858, *Multiprotocol Extensions for BGP-4*, June 2000

- Route refresh capability:

  E. Chen, RFC 2918, *Route Refresh Capability for BGP-4*, September 2000

- Outbound route filtering (ORF) capability:

  E. Chen, Y. Rekhter, RFC 5291, *Outbound Route Filtering Capability for BGP-4*, August 2008

- Address prefix-based ORF capability:

  E. Chen, S. Sangli, RFC 5292, *Address-Prefix-Based Outbound Route Filter for BGP-4*, August 2008

- Graceful restart capability:

  S. Sangli, Y. Rekhter, R. Fernando, J. Scudder, E. Chen, RFC 4274, *Graceful Restart Mechanism for BGP*, January 2007

- Four-byte autonomous system (AS) capability:

  Q. Vohra, E. Chen, Internet Draft, *BGP Support For Four-Octet AS Number Space*, draft-ietf-idr-as4bytes-03.txt, May 2001

The following additional features are also supported:

- Routing policies, including these types of filters:

  – Prefix lists

  – AS path lists

  - Route maps

- Address family identifier (AFI) and subsequent address family identifier (SAFI) configuration, including:
  - IPv4 unicast
  - IPv4 labeled
  - IPv4 multicast
  - IPv4 VPN
  - IPv6 unicast
  - IPv6 labeled
  - IPv6 VPN
- BGP route sourcing, including these methods:
  - Redistribution from other routing protocols into the BGP routing domain
  - Origination of BGP routes through the **`network`** command in BGP address family configuration mode

    Note: The **`network`** command is available in the local context only. You cannot configure the network statement inside an IP VPN.
- Route aggregation through the support of the AS_SET attribute
- Default origination—both conditional and unconditional
- Maximum number of prefixes setting
- Next-Hop-Triggered BGP best-path calculation
- Multipath capability for both internal BGP (iBGP) and external BGP (eBGP).
- Peer groups, including these features:
  - Address family-specific grouping
  - Decoupling of peer groups and default origination
- 6PE: IPv6 on the provider edge (PE), which allows IPv6 to be tunneled over a MPLS/IPv4-based core network.
- 6VPE: VPN IPv6 on the provider edge (PE), which allows VPN IPv6 to be tunneled over a MPLS/IPv4-based core network.
- Route-flap statistics for both iBGP and eBGP
- Fast-reset of BGP sessions on receipt of a link-failure event.

- Configurable Minimum Route Advertisement Interval (MRAI) (using the *advertisement-interval* command).
- Optional verification of the first AS number in a received AS path from an eBGP peer.
- Accounting of routes by these methods:
  - Number of routes sourced
  - Number of routes accepted, active, dampened, and historical from each peer
  - Number of routes advertised to a peer
- Advanced debug facilities, including these features:
  - Per-neighbor based generation of debug messages
  - Storage and display of malformed messages and notification messages
  - Peer reset history

In-depth information on how BGP is structured, and how it operates, is described the sections that follow.

1.1 Introduction to iBGP and eBGP

Routers that belong to the same AS and exchange BGP updates are running iBGP, and routers that belong to different autonomous systems and exchange BGP updates are running eBGP.

Figure 1 illustrates the concept of autonomous systems and iBGP versus eBGP.

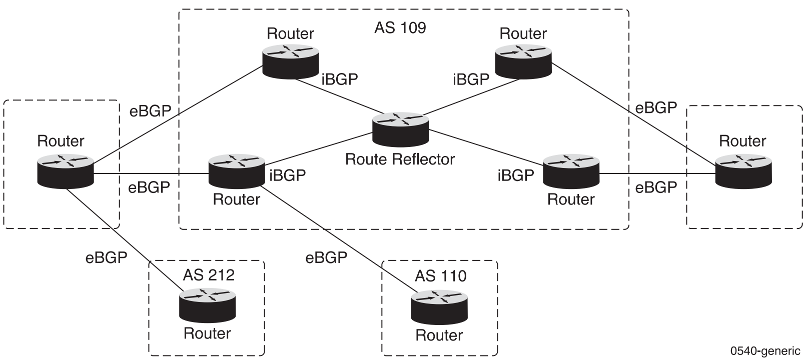

Figure 1 Autonomous Systems and iBGP Versus eBGP Networks

1.2 Introduction to iBGP Route Reflectors

Typically, iBGP speakers must be fully meshed. Any BGP speaker that receives messages from an external router must advertise the routes it receives to all BGP speakers in its autonomous system. However, if a route reflector is configured, although it must have connections to all other BGP speakers in the AS, not all other BGP speakers must be fully meshed. When a BGP speaker in the AS receives messages from an external router, it is sufficient to advertise these routes only to the route reflector, which then readvertises the bestpath of each route to all other BGP speakers in the AS.

Internal peers of the route reflector are divided into two groups: client peers and nonclient peers. A route reflector reflects routes between these two groups. The route reflector and its client peers form a cluster. Nonclient peers must be fully meshed with each other. Client peers are not required to be fully meshed and do not communicate with BGP speakers outside their cluster. If it is required, peer client-to-peer client route reflection can be disabled.

When the route reflector receives an advertised route:

- Any route from an external BGP speaker is advertised to all peers.
- Any route from a nonclient peer is advertised to all client peers.
- Any route from a client peer is advertised to all peers.

Figure 2 shows an example of iBGP networking using route reflection.

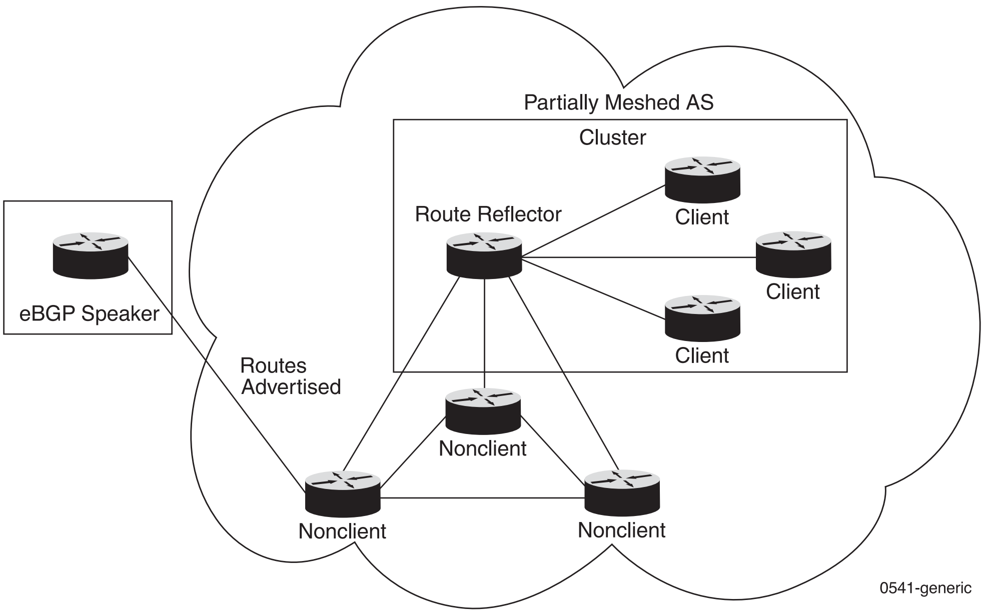

Figure 2 Example of an iBGP Network Using Route Reflection

1.3 Introduction to iBGP Confederations

Another way to reduce iBGP mesh is to divide an autonomous system into subautonomous systems grouped by a routing domain identifier. The AS and its subautonomous systems are part of the same confederation. Externally, the confederation looks like a single AS. Each subautonomous system is fully meshed within itself and has a few connections to other subautonomous systems in the confederation.

Neighbors from other subautonomous systems are treated as special eBGP peers. Even though peers in different subautonomous systems engage in eBGP sessions, they exchange routing information as if they were iBGP peers. Specifically, the next-hop, the Multi-Exit Discriminator (MED), and local preference information is preserved, so that a single Interior Gateway Protocol (IGP) is used for all of the subautonomous systems; see Figure 3.

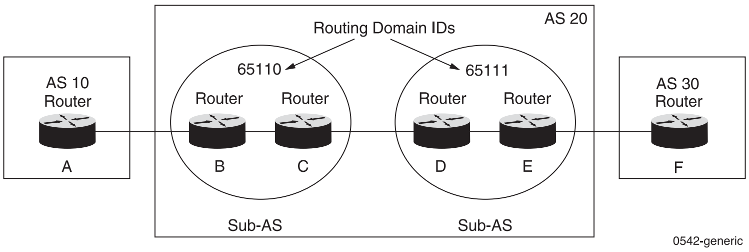

Figure 3 Example of an iBGP Confederation

Note: For iBGP paths, the MED is always 0.

1.4 Route Aggregation

BGP4 supports Classless InterDomain Routing (CIDR). With CIDR, routers use the network prefix to determine the dividing point between the network number and the host number. For example, the range of addresses 128.186.1.0 to 128.186.1.255 can be represented as the network prefix 128.186.1.0/24; the 24 indicates that all addresses in the segment agree in their first 24 bits.

In addition, CIDR does not require a network to be of standard size, as is the case in classful addressing, which provides 8-bit (Class A), 16-bit (Class B), and 24-bit (Class C) network deployment. This flexibility in CIDR enables the creation of arbitrarily sized networks.

Of particular importance is CIDR's ability to lend itself to the concept of route aggregation. The Internet is divided into addressing domains. Within a domain, detailed information is available about all of the networks that reside in the domain. Outside of an addressing domain, however, only the common network prefix is advertised. By allowing a single routing table entry to specify a route to many individual network addresses, aggregation minimizes the size of the routing table. A router cannot aggregate an address if it does not have a more specific route of that address in the BGP routing table. More specific routes can be injected in the BGP routing table by incoming updates from other autonomous systems.

1.5 Next-Hop-Triggered BGP Best-Path Calculation

By default, a change in a next-hop reachability does not immediately trigger a BGP best-path calculation. Instead, the SmartEdge router periodically checks whether any next-hop has changed since the last check, and if there has been a change, it runs BGP best-path calculations for all routes. This behavior reduces computational burden, but delays network convergence.

The next-hop-triggered BGP best-path calculation feature allows the user to change this default behavior. When the feature is enabled, the SmartEdge router runs the best-path calculation immediately upon notification of a next-hop change by the RIB, thereby improving the convergence time. A hold down time and a backoff mechanism prevent unnecessary churn and excessive CPU usage during network instability.

If next-hop best-path calculation is enabled for an address family, the periodic next-hop check is not performed for that address family.

Use the following commands to configure next-hop-triggered BGP best-path calculation:

- **`nexthop triggered`**—Enables the triggering of immediate BGP best-path calculation on notification of a next-hop withdrawal by the RIB.

- **`nexthop triggered delay`**—Defines the delay, in seconds or milliseconds, before starting the best-path calculation after a next-hop change notification. This delay allows time for the accumulation of more than one next-hop change into a single best-path calculation when multiple next-hop change events are expected in response to a network event.

- **`nexthop triggered holdtime`**—Defines the minimum interval, in seconds or milliseconds, between two consecutive next-hop triggered best-path calculations. During IGP churn, BGP limits the next-hop triggered best-path calculations first by the configured hold time, then by increasing the time through the configured backoff value for every new next-hop change that occurs before the expiration of the hold time. A hold time is not applicable across different next hops. In other words, a next-hop change for 10.12.13.14/32 followed by a next-hop change for 10.40.50.60/32 does not trigger a hold time. A next-hop change caused by an IGP route update and a next-hop change caused by an LSP route update will have individual hold times.

Note: Next-hop-triggered BGP best-path calculation is not supported for the IPv4 multicast address family.

The following commands show information related to BGP next-hop scanning:

- **`show bgp route summary [detail]`**

- **`debug bgp event`**

1.5.1 BGP Multipath

By default, BGP multipath capabilities are disabled, which means BGP installs a single path in the RIB for each destination. If that path fails and no other path has installed a path for that prefix, traffic destined for that path is lost until the path is available again.

When BGP multipath is enabled with the *multi-paths* command, BGP installs multiple best equal-cost paths in the routing table for load-balancing traffic to BGP destinations. With multipath, the paths can be:

- All iBGP (configured with the **`multi-paths internal`** ***`path-num`*** command)
- All eBGP (configured with the **`multi-paths external`** ***`path-num`*** command)
- In the VPN context, a combination of iBGP and eBGP, where only 1 eBGP path is allowed, and the number of allowed iBGP equal-cost paths is equal to the maximum number of paths allowed (configured with the **`multi-paths eibgp`** ***`path-num`*** command) minus 1. For example, if you configure **`eibgp 7`**, 6 iBGP paths and 1 eBGP path are installed in the RIB.

  Note: The **`eibgp`** keyword is not supported for IPv6 traffic.

When BGP multipath capabilities are enabled, even though multiple paths are installed in the RIB, BGP advertises only one path (the BGP best path) to its peers.

1.6 MP-BGP

Multiprotocol BGP (MP-BGP) makes use of multiprotocol extensions to BGP4, as defined in RFC 2283, *Multiprotocol Extensions for BGP-4*, that allow other protocols to use BGP to exchange protocol-specific information.

One of the main advantages of MP-BGP is the ability to use BGP's scalability and policy control, to easily configure routers to peer with other interdomain routers, exchange multicast source route information, and configure multicast routing policies using familiar BGP commands. MP-BGP also carries two sets of routes: one set for unicast routing and one set for multicast routing, allowing you to configure separate routing policies for unicast and multicast routes.

1.7 Routing Policy Triggered Update

Before Release 2.5, whenever there was a change in an inbound or outbound routing policy, such as a prefix-list, as-path-list, or route-map, for a BGP peer, the **`clear bgp neighbor`** ***`ip-addr`*** **`soft`** [**`in`** | **`out`**] command had to be manually issued to make the policy change effective. Currently, routing policy changes automatically take effect, and issuing the **`clear bgp neighbor`** *ip-addr* **soft** [**in** | **out**] command to update routing policies can cause updates to be unnecessarily sent, so it is not recommended.

To aggregate multiple policy changes, the operating system performs the necessary action 15 seconds after a policy change.

Caution!

Risk of dropped connection. If the remote peer does not support the BGP Route Refresh Capability, an inbound policy change for the peer results in an automatic hard reset of the session. To reduce the risk, ensure that the remote peer supports the BGP Route Refresh Capability.

1.8 Non-Intrusive MD5 Password Change

The non-intrusive Message Digest 5 (MD5) password change feature for BGP allows you to change the password for a BGP peer without resetting the BGP session. The sections that follow describe in detail how the non-intrusive MD5 password change feature is implemented.

1.8.1 Replace a Password

When an old MD5 password is replaced by a new one in a BGP peer configuration, both passwords are allowed to coexist for authentication until the old password expires. To facilitate a smooth transition from the old to new password, a new configuration can be used to specify the time interval during which the old MD5 password coexists with the new one.

For a TCP connection that is already established, or is in one of the closing states when an existing password is replaced by a new MD5 password, both password strings coexist for authentication during the specified time interval before the old MD5 password expires. The old MD5 password continues to be used for authentication until either the password expires, or the remote TCP for the peer uses a new MD5 password.

For a TCP connection that is not yet established, when the old password is replaced, the local TCP immediately uses the new MD5 password.

Note: BGP keeps only the latest password string configured and the previous password to be replaced. That is, if a third password is configured before the timer for first (active) password expires, the oldest password is immediately deleted, and the expiration timer is started for the second password.

1.8.2 Add a New Password

This feature does not apply when configuring a new MD5 password for a peer while there is no existing password already configured for the peer. The BGP peer session is reset after the new MD5 password is configured.

1.8.3 Delete a Password

This feature does not apply when explicitly deleting a MD5 password from the BGP peer configuration.

When the current active MD5 password is deleted from the configuration, the old password (if existing) and the current password are both immediately deleted, and the BGP session with the peer is reset.

Note: To avoid BGP sessions from being reset when changing a peer MD5 password, we recommend that you do not delete the password from the configuration, and always use the **`password`** command to implicitly replace the password.

1.9 BGP Prefix-Based Outbound Route Filtering

A BGP speaker can use its local routing policy to filter out unwanted routes received from peers of the speaker. However, filtering uses resources on both the sender and receiver, which must generate and process BGP updates for the unwanted routes. To preserve resources in your network, you can use BGP prefix-based outbound route filtering (ORF) to prevent the generation and processing of these BGP updates.

With BGP prefix-based ORF, a BGP speaker sends a set of outbound route filters to a BGP peer. The peer applies these filters in addition to any locally configured outbound filters. These filters prevent unnecessary outbound routing updates from being sent to the speaker.

To configure ORF on your system:

- Configure the sending BGP speaker to send ORFs:
  - Use the *send filter prefix-list* command to advertise to a BGP peer that this BGP speaker can send prefixed-based filtering to the peer.
  - Use the *prefix-list* ***`pl-name`*** **`in`** command to apply the IP prefix list to a neighbor address family. Replace the ***`pl-name`*** argument with the name of the prefix list you want to apply.
- On the receiving BGP speaker (the speaker that applies the ORFs), use the *accept filter prefix-list* command to configure the receiving BGP speaker to accept ORFs received from the sending BGP speaker.

Note: When you enter the *accept filter prefix-list* and *send filter prefix-list* commands, the connection between the BGP speakers automatically resets. Because ORF capabilities are communicated between BGP speakers during BGP connection establishment, the *accept filter prefix-list* and *send filter prefix-list* commands do not take effect until the BGP connection resets.

1.10 BGP Graceful Restart Capabilities

Graceful restart is enabled on all BGP routing instances. When configured as a BGP speaker, the router:

- Preserves the forwarding state of the BGP speaker during a BGP restart
- Generates the end-of-Routing Information Base (RIB) marker on the completion of initial routing updates

The BGP speaker advertises these graceful restart capabilities to the peers.

Keep the following in mind when configuring graceful restart for a BGP routing instance:

- Graceful restart is always enabled on all BGP routing instances.
- Graceful restart is supported for all IPv4 and IPv6 address families. You must use the *send label* command to enable the negotiation of IPv4 and IPv6 labeled address families.
- When an iBGP peer restarts, the restarting and helper peers exchange graceful restart capabilities. In addition, all iBGP peers within the same domain exchange their graceful restart capabilities, including the list of IP address families with routes that can be gracefully restarted. The helper router helps restart only those iBGP peers that have the same address-family capabilities. Use the following commands to configure address-family capabilities:
  - See *address-family ipv4 (BGP)*.
  - See *address-family ipv6 unicast*.
  - See *address-family ipv6 vpn*.

Note: To ensure that routes are maintained in an iBGP system, we recommend configuring all iBGP peers in a domain with the same address-family capabilities.

Page 12 illustrates an example of how routes are maintained in an iBGP system. This example shows a system of four iBGP peers, where router A is the helper router. If router D fails, router A retains the address families and routes from router D only if routers B and C are configured with the same routing capabilities as router D. For example, if router D is configured with IPv4 unicast address family capabilities, router A retains those IPv4 unicast routes only if routers B and C are also configured with IPv4 unicast address family capabilities. If router

D is configured with IPv4 unicast and IPv6 unicast capabilities, but routers B and C are configured only with IPv4 unicast address family capabilities, Router A retains only the IPv4 unicast address families and routers when router D fails.

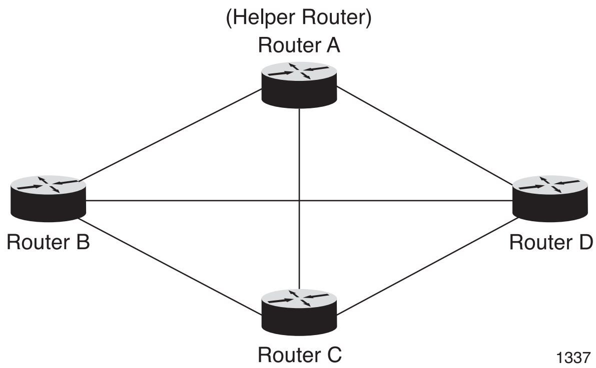

Figure 4 iBGP Graceful Restart: How iBGP Routes Are Maintained

1.11 Fast-Reset of BGP Sessions

The BGP fast-reset feature enables fast resetting of a BGP peer session when the links used to reach a neighbor go down.

When fast-reset is disabled, the BGP session is not reset immediately when the link used to reach that peer goes down. Instead, the BGP sessions remain connected to the peer until the configured BGP holdtime timer (set with the **`timers keepalive`** command in BGP router configuration mode) expires. If no packets are received from a peer within the configured hold time, the BGP session is reset.

When fast-reset is enabled, the configured hold time is ignored, and BGP drops its session with a peer immediately if the link to that peer goes down. When BGP fast-reset is enabled on peers, packet loss is minimized during link failures.

Fast-reset is supported on directly connected and multihop BGP sessions.

For directly connected eBGP peers, use the *fast-reset* command in BGP router configuration mode to specify the amount of time (in seconds) that must pass before the BGP routing process drops sessions of directly connected external peers when the link used to reach them goes down. In this case, the fast-reset configuration applies to all eBGP peers that are directly connected to the local system.

For iBGP or multihop eBGP sessions, a peer is reachable through a number of interfaces. To configure fast-reset for iBGP or multihop eBGP sessions, do the following:

1 Use the *fast-reset* command in BGP neighbor configuration or BGP peer group configuration mode to do the following:

- Access BGP neighbor fast-reset configuration mode, where you create a BGP fast-reset interface list of links to a BGP neighbor; BGP fast-reset is triggered when all links in the list go down.

- Configure the interval that must pass before BGP routing process triggers fast-reset after all of the links in the BGP fast-reset interface list go down.

2. Use the *interface* command in BGP neighbor fast-reset configuration mode to add an interface to a BGP fast-reset interface list; you can add up to 10 interfaces.

Consider the following when configuring BGP fast-reset on a multihop BGP session:

- A BGP session remains active as long as at least one of the interfaces in the BGP fast-reset interface list is up. When all of the interfaces in the list go down, BGP ignores the configured hold time (specified by the **`timers keepalive`** command) and, instead, waits for the specified fast-reset ***`interval`*** before removing its sessions with the affected neighbor.

- If none of the specified interfaces are up in the configured BGP fast-reset interface list, BGP does not establish a session with a neighbor (regardless of whether the neighbor is reachable).

- If all of the interfaces configured in a fast-reset interface list go down, the BGP session goes down and does not become active again until at least one of the down interfaces comes up.

- When configuring BGP fast-reset for BGP peer groups:

  - The BGP fast-reset configuration for a particular neighbor takes precedence over the BGP fast-reset configuration for a peer group. For example, if a BGP neighbor is configured with a fast-reset interval of 50 milliseconds, and that neighbor belongs to a peer group that is configured with a fast-reset interval of 20 seconds, the BGP neighbor ignores the peer group configuration and uses the 50-millisecond interval.

  - If a neighbor does not already have BGP fast reset configured, that neighbor inherits the fast-reset configuration from the peer group.

  - If a neighbor has its own BGP fast-reset configuration, to return that neighbor to the default (where the neighbor inherits the BGP fast-reset configuration from the peer group), you must remove the neighbor from and the peer group and then add it back.

1.12 BGP Minimum Route Advertisement Interval

You can configure the Minimum Route Advertisement Interval (MRAI) by using the *advertisement-interval* command. The MRAI starts when an UPDATE message is sent to a BGP neighbor or peer group. After sending an UPDATE message to the specified BGP peers, the router waits for the specified MRAI before sending the next UPDATE message. If a route change occurs and the MRAI has passed since the last UPDATE message was sent, the router immediately sends an UPDATE message to the peer. If a route change occurs and the MRAI has not passed since the last UPDATE message was sent to the peer, the router waits the specified MRAI before sending a new UPDATE message.

Page 14 illustrates an example of how MRAI sends UPDATE messages to BGP peers.

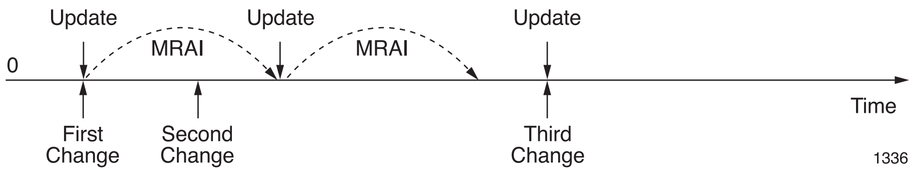

Figure 5 MRAI Example

If the MRAI in Page 14 is set to 20 seconds:

- The first route change occurs, so the BGP router immediately sends an UPDATE message to the specified peers and starts the MRAI. (The first route change always occurs at 0 seconds according to the MRAI).
- A second route change occurs 2 seconds after the MRAI starts. In this case, the router waits 18 more seconds before sending an UPDATE message to the specified peers. The router restarts the MRAI after sending the UPDATE message to the specified peers.
- A third route change occurs at 60 seconds. Because 40 seconds passed since the last UPDATE message was sent (40 seconds is greater than the configured MRAI of 20 seconds), the router immediately sends the third UPDATE message.

2 Configuration and Operations Tasks

Note: In this section, the command syntax in the task tables displays only the root command.

To configure BGP, perform the tasks described in the sections that follow.

2.1 Configuring BGP Routing Instances and Instance Attributes

A BGP routing instance enables the SmartEdge router to be a BGP speaker. In addition, many BGP parameters that can affect the global routing process can be configured within a BGP routing instance.

To configure a BGP routing instance and other instance attributes, perform the tasks described in the sections that follow.

2.1.1 Creating and Configuring a BGP Routing Instance

To configure a BGP routing instance, perform the tasks described in Table 1.

Table 1 Configure a BGP Routing Instance

Task	Root Command	Notes
Create a BGP routing instance using an autonomous system number (ASN) and enter BGP router configuration mode.	*router bgp*	Enter this command in context configuration mode.
Allow the comparison of the Multi-Exit Discriminator (MED) for paths from all BGP neighbors in different autonomous systems.	*bestpath med always-compare*	By default, the MED comparison is done by the BGP routing instance on BGP paths received from BGP neighbors in one other autonomous system. When enabled, this command changes the default behavior by allowing comparison of MEDs among paths regardless of the autonomous system from which the paths are received. For iBGP paths, the MED is always 0.

Table 1 Configure a BGP Routing Instance

Task	Root Command	Notes
Disable (or reenable, if disabled) the verification of the first AS number in a received AS path from an eBGP peer.	*[no] enforce first-as*	By default, a BGP router compares the remote AS number of an eBGP peer with the first AS number in the paths received from that peer. If those AS numbers do not match, the BGP router: • Sends a NOTIFICATION message to eBGP peer that contains error code 3 (UPDATE Message Error) and error subcode 11 (Malformed AS_PATH). For more information on these error codes, see RFC 4271, *A Border Gateway Protocol 4 (BGP-4)*. • Drops the session with the eBGP peer. Use the **`no enforce first-as`** command to disable this feature.
Specify a period of time that must pass before the BGP routing process drops sessions of directly connected external peers once the link used to reach them goes down.	*fast-reset*	By default, BGP sessions remain connected after the outbound interface goes down. BGP sessions are dropped after the BGP hold time value, set with the **`timers keepalive`** command in BGP router configuration mode, is exceeded.
Enable or disable graceful restart for a BGP routing instance.	*[no] graceful-restart*	Graceful-restart is enabled on all BGP routing instances by default.
Configure the local preference attribute for the BGP routes.	*local-preference*	The local preference value is applied to BGP routes that do not have the local-preference attribute assigned to them.

Table 1 Configure a BGP Routing Instance

Task	Root Command	Notes
Log BGP neighbor resets.	*log-neighbor-changes*	—
Enable multipath capabilities on a system, so that multiple paths to the same destination are installed in the routing table.	*multi-paths*	Although multiple paths are installed, only one path (the best path available) is advertised. Use the **`external`** and **`internal`** keywords to enable eBGP and iBGP equal-cost paths, respectively. Use the **`eibgp`** keyword to enable multipath load balancing using both eBGP and iBGP paths in a BGP/MPLS VPN. This keyword is not supported for IPv6 traffic.
Configure a fixed BGP router ID.	*router-id (BGP)*	By default, the BGP router ID is the IP address of a loopback interface if one is configured. If a loopback interface is not configured, the interface with the highest IP address is used as the router ID. Peering sessions are reset when the router ID is changed. If a context does not contain any IPv4 address configuration and BGP is being used, you must configure the **`router-id`** command in the context or routing protocol instance level. If you configure a context with only IPv6 addresses and no IPv4 addresses and run BGP in that context, BGP does not establish a relationship with any neighbors if the **`router-id`** command is not configured.

Table 1 Configure a BGP Routing Instance

Task	Root Command	Notes
Configure the time interval, in seconds, during which an old MD5 password can coexist with a new MD5 password for authentication.	*timers password*	Configuring the password timer interval affects only the BGP peers which have existing MD5 passwords replaced after this configuration is committed.
Modify keepalive and holdtime timers for all BGP neighbors.	*timers keepalive*	By default, the keepalive timer is set to 60 seconds and the holdtime value is set to 180 seconds.
Configure IPv4 multicast or unicast address family attributes.	—	For the complete list of tasks used to configure IPv4 address family attributes, see Configure IPv4 Address Family Attributes for a BGP Routing Instance.
Configure IPv6 unicast address family attributes.	—	For the complete list of tasks used to configure IPv6 address family attributes, see Configure IPv6 Address Family Attributes for a BGP Routing Instance.
Configure the BGP graceful restart characteristics.	—	For the complete list of tasks used to configure BGP graceful restart, see Configure Graceful Restart Characteristics for a BGP Routing Instance.
Configure BGP Route Reflection.	—	For the complete list of tasks used to configure BGP route reflection, see Configure BGP Route Reflection.
Configure BGP confederations.	—	For the complete list of tasks used to configure BGP confederations, see *Configuring BGP Attribute-Based Accounting* in *Configuring Routing Policies*.

2.1.2 Configuring IPv4 Address Family Attributes for a BGP Routing Instance

To configure the IPv4 address family attributes for a BGP routing instance, perform the tasks described in Table 2.

Table 2 Configure IPv4 Address Family Attributes for a BGP Routing Instance

Task	Root Command	Notes
Specify the use of standard IP Version 4 (IPv4) multicast or unicast address prefixes for the BGP routing instance, and access BGP address family configuration mode.	**`address-family ipv4`** command See *address-family ipv4 (BGP)*.	Enter this command in BGP router configuration mode. Include the **`multicast`** keyword to specify a multicast address prefix, or the **`unicast`** keyword to specify a unicast address prefix.
Create an aggregate entry in the BGP database for the BGP address family.	*aggregate-address*	—
Enable eBGP route dampening for the specified BGP address family.	*dampening*	—
Configure the administrative distance values for a BGP address family.	*distance (BGP address family)*	When installing and advertising the best path, the BGP best-path algorithm uses the administrative distance value in instances where multiple paths are available. If the distance of a path is greater than the maximum allowable distance out of the distances configured for iBGP, eBGP, and local, BGP removes that path from the list of best-path candidates. If only one path is up, BGP installs that path and ignores the distance value.
Enable route-flap statistics accounting for the BGP address family.	*flap-statistics*	—
Originate BGP routes that are advertised to peers.	*spf-timers*	—

Table 2 Configure IPv4 Address Family Attributes for a BGP Routing Instance

Task	Root Command	Notes
Redistribute routes learned through other protocols into the BGP routing process.	*redistribute (BGP)*	Be aware that the *redistribute (BGP)* command is available in BGP address family configuration mode for unicast address prefixes only. See Example: Configure BGP Route Redistribution and Aggregation for an example of how to configure route redistribution and aggregation for a BGP instance.
Assign a traffic index to routes installed for a BGP address family.	*table-map*	Traffic index counters are maintained on interfaces with traffic index accounting enabled. For more information about BGP attribute-based accounting, see the *Configuring BGP Attribute-Based Accounting* section in *Configuring Routing Policies*.
Enable the triggering of immediate BGP best-path calculation on notification of a next-hop withdrawal by the RIB, and configure next-hop scan parameters.	*nexthop triggered*	The **`nexthop triggered`** command is not supported for the IPv4 multicast address family.
Define the delay before starting the best-path calculation after a next-hop change notification.	*nexthop triggered delay*	This delay allows the accumulation of more than one next-hop change into a single best-path calculation when multiple next-hop changes events are expected in response to a network event.
Define the minimum interval between two consecutive next-hop triggered best-path calculations.	*nexthop triggered holdtime*	You must enter this command separately for each BGP instance for which you want to define the minimum interval between next-hop triggered best-path calculations.

2.1.3 Configuring IPv6 Address Family Attributes for a BGP Routing Instance

To configure the IPv6 address family attributes for a BGP routing instance, perform the tasks described in Table 3.

Table 3 Configure IPv6 Address Family Attributes for a BGP Routing Instance

Task	Root Command	Notes
Specify the use of standard IP Version 6 (IPv6) unicast address prefixes for the BGP routing instance, and access BGP address family configuration mode.	*address-family ipv6 unicast*	Enter this command in BGP router configuration mode.
Create an aggregate entry in the BGP database for the BGP address family.	*aggregate-address*	—
Enable eBGP route dampening for the specified BGP address family.	*dampening*	—
Configure the administrative distance values for a BGP address family.	*distance (BGP address family)*	When installing and advertising the best path, the BGP best-path algorithm uses the administrative distance value in instances where multiple paths are available. If the distance of a path is greater than the maximum allowable distance out of the distances configured for iBGP, eBGP, and local, BGP removes that path from the list of best-path candidates. If only one path is up, BGP installs that path and ignores the distance value.
Enable route-flap statistics accounting for the BGP address family.	*flap-statistics*	—
Originate BGP routes that are advertised to peers.	*spf-timers*	—
Enable the triggering of immediate BGP best-path calculation on notification of a next-hop withdrawal by the RIB, and configure next-hop scan parameters.	*nexthop triggered*	—
Define the delay before starting the best-path calculation after a next-hop change notification.	*nexthop triggered delay*	This delay allows the accumulation of more than one next-hop change into a single best-path calculation when multiple next-hop changes events are expected in response to a network event.

Table 3 Configure IPv6 Address Family Attributes for a BGP Routing Instance

Task	Root Command	Notes
Define the minimum interval between two consecutive next-hop triggered best-path calculations.	*nexthop triggered holdtime*	You must enter this command separately for each BGP instance for which you want to define the minimum interval between next-hop triggered best-path calculations.
Redistribute routes learned through other protocols into the BGP routing process.	*redistribute (BGP)*	See Example: Configure BGP Route Redistribution and Aggregation for an example of how to configure route redistribution and aggregation for a BGP instance.
Assign a traffic index to routes installed for a BGP address family.	*table-map*	Traffic index counters are maintained on interfaces with traffic index accounting enabled. For more information about BGP attribute-based accounting, see *Configuring BGP Attribute-Based Accounting* in *Configuring Routing Policies*.

2.1.4 Configuring Graceful Restart Characteristics for a BGP Routing Instance

Graceful restart is always enabled in all BGP routing instances, and is supported for both IPv4 and IPv6 address families. You cannot disable BGP graceful restart on a BGP neighbor, but you can configure some characteristics.

Note: Before you can configure graceful restart for a BGP routing instance, you need to create and configure a BGP routing instance, as described in Creating and Configuring a BGP Routing Instance.

To configure the graceful restart characteristics for a BGP routing instance, perform the tasks described in Table 4. Enter all commands in BGP router configuration mode.

Table 4 Configure Graceful Restart Characteristics for a BGP Routing Instance

Task	Root Command	Notes
Set the maximum amount of time that it will take for a local BGP peer to come up after it has been reset.	*maximum restart-time*	—

Table 4 Configure Graceful Restart Characteristics for a BGP Routing Instance

Task	Root Command	Notes
Set the maximum amount of time the local BGP speaker retains routes it has previously received from a remote peer once that remote peer restarts the connection.	*maximum retain-time*	Any routes that have not been updated by the remote peer are deleted by the local peer after the local peer receives the end-of-RIB marker from the remote peer, or after the timer expires.
Set the maximum delay time for the BGP routing process after a reset has occurred before performing initial best path calculations.	*maximum update-delay*	Use this feature when all peers do not support a graceful restart, or when a peer may not send an end-of-RIB marker.

2.1.5 Configuring BGP Route Reflection

If a BGP route reflector is configured, while it must have connections to all other BGP speakers in the AS, not all other BGP speakers must be fully meshed. When a BGP speaker in the AS receives messages from an external router, it is sufficient to advertise these routes only to the router reflector, which then readvertises the routes to all other BGP speakers in the AS.

Note: Before you can configure a BGP router reflector, you need to create and configure a BGP routing instance, as described in Creating and Configuring a BGP Routing Instance.

To configure BGP route reflection, perform the tasks described in Table 5. Enter all commands in BGP router configuration mode.

Table 5 Configure BGP Route Reflection

Task	Root Command	Notes
Enable client-to-client reflection.	*client-to-client reflection*	By default, routes are reflected between clients of a route reflector.
Disable client-to-client reflection.	*no client-to-client reflection*	Disable client-to-client reflection when you do not want routes that have been learned from one client to be reflected to other clients; for example, when clients are fully meshed.
Assign a separate cluster ID to each route reflector.	*cluster-id*	Use this command when there is more than one route reflector in a cluster.

2.1.6 Configuring a BGP Confederation

To reduce iBGP mesh, you can divide an autonomous system into subautonomous systems grouped by a routing domain identifier. The AS and its subautonomous systems are part of a BGP confederation. Externally, the confederation looks like a single autonomous system.

Note: Before you can configure a BGP confederation, you need to create and configure a BGP routing instance, as described in the Creating and Configuring a BGP Routing Instance.

To configure a BGP confederation, perform the tasks described in Table 6. Enter all commands in BGP router configuration mode.

Table 6 Configure a BGP Confederation

Task	Root Command	Notes
Configure a BGP confederation.	*confederation identifier*	—
Configure the subautonomous systems that belong to the BGP confederation.	*confederation peers*	—

2.2 Configuring BGP Neighbors and Neighbor Attributes

BGP speakers (BGP-enabled routers) that exchange inter-AS routing information are called BGP neighbors. BGP supports two kinds of neighbors: internal and external. Internal neighbors are in the same AS; external neighbors are in different autonomous systems. External neighbors must be adjacent to each other and share the same subnet, while internal neighbors may be located anywhere inside the same autonomous system.

To enable BGP speakers to effectively communicate with each other, each BGP speaker must be configured with information about its BGP neighbors.

The sections that follow describe how to configure a BGP neighbor and other neighbor attributes.

2.2.1 Configuring a BGP Neighbor

To configure a BGP neighbor, perform the tasks described in Table 7.

Note: Before you can configure a BGP neighbor, you need to create and configure a BGP routing instance, as described in Creating and Configuring a BGP Routing Instance.

Table 7 Configure a BGP Neighbor

Task	Root Command	Notes
Enter BGP router configuration mode.	*router bgp*	Enter this command in context configuration mode.
Create a BGP neighbor and access BGP neighbor configuration mode.	*neighbor (BGP)*	Enter this command in BGP router configuration mode. To configure an external GRP (eBGP) neighbor, include the **`external`** keyword in the neighbor command string. To configure an internal GRP (iBGP) neighbor, include the **`internal`** keyword in the neighbor command string.
Advertise to a peer that this BGP speaker is willing to accept address prefix-based route filtering from the peer.	*accept filter prefix-list*	Because ORF capabilities are communicated between BGP speakers during BGP connection establishment, the *accept filter prefix-list* command does not take effect until the BGP connection is reset.
Modify the MRAI (minimal interval at which BGP routing updates are sent to the specified neighbor).	*advertisement-interval*	Range is from 0 to 600. For external BGP (eBGP), the default value is 30. For internal BGP (iBGP), the default value is 5.
Enable Bidirectional Forwarding Detection (BFD) for an eBGP neighbor.	*bfd*	BFD is a simple Hello protocol that provides the ability to detect communication failures in less than one second. When BFD detects a communication failure to the eBGP neighbor, the neighbor is reset. BFD can be enabled only for eBGP neighbors; enabling BFD for an iBGP neighbor generates an error message. For more information about BFD, see *Configuring BFD*.
Associate a description with the neighbor.	*description (BGP)*	—
Configure the maximum number of hops used to reach an eBGP neighbor when the neighbor is not directly connected.	*ebgp-multihop*	This command must be enabled for BGP connections to be established with neighbors that are not directly connected.

Table 7 Configure a BGP Neighbor

Task	Root Command	Notes
Enable the BGP time-to-live (TTL) security check in the kernel for the BGP neighbor.	*enforce ttl*	For the BGP TTL security check to function correctly, it must be enabled on both ends of an eBGP session. Enabling only one end causes the eBGP session to drop.
Configure the ASN that the BGP routing process uses to peer with the specified eBGP neighbor.	*local-as*	This command supports the **`no-prepend`** option to disable prepending the local AS to inbound route updates received from the eBGP neighbor and the **`replace-as`** option to replace the global ASN with the local AS in the outbound message.
Advertise the local peer address as the next-hop address.	*next-hop-self*	By default, when a BGP neighbor receives BGP routes from an eBGP neighbor, routes are sent to iBGP neighbors without changing the next-hop address.
Configure an encrypted MD5 password for the BGP neighbor.	*password (BGP)*	—

Table 7 Configure a BGP Neighbor

Task	Root Command	Notes
Apply the attributes of a configured BGP peer group to one or more BGP neighbors.	*peer-group*	You can assign a neighbor to a peer group only if the neighbor and the peer group are of the same type—external or internal BGP. If a neighbor belongs to a particular peer group, you cannot configure it to belong to another peer group. You must first explicitly delete the previous peer group membership before reconfiguring the peer membership. Attributes are inherited from the peer group to which a neighbor is assigned. The following BGP neighbor configuration mode commands represent attributes that you cannot customize per neighbor when the neighbor is assigned to a peer group: **`advertisement-interval`**, **`ebgp-multihop`**, **`local-as`**, **`send community`**, and **`timers keepalive`**. Attributes inherited from a peer group that you can customize per neighbor include those set by the following commands: **`description`**, **`password`**, **`send prefix`**, **`shutdown`**, and **`update-source`**.
Configure the ASN of the eBGP neighbor.	*remote-as*	—
Send the community attribute to the specified eBGP neighbor.	*send community*	—
Advertise to a BGP peer that this BGP speaker would like to send prefixed-based filtering to the peer.	*send filter prefix-list*	Because ORF capabilities are communicated between BGP speakers during BGP connection establishment, the *send filter prefix-list* command does not take effect until the BGP connection is reset.
Administratively shut down a BGP session with the specified neighbor.	*shutdown (BGP)*	This command temporarily shuts down a BGP session without removing a BGP neighbor from the configuration.

Table 7 Configure a BGP Neighbor

Task	Root Command	Notes
Specifies the interval the router waits for a BGP peer to come up after a graceful restart before starting the best path computation.	*timers active-open*	Range is from 1 through 600 seconds. Be aware that the **`timers active-open`** interval configuration in BGP neighbor configuration mode takes precedence over the **`timers active-open`** interval configuration in BGP peer group configuration mode.
Modify keepalive and holdtime timers for a specific neighbor.	*timers keepalive*	Values set for a BGP neighbor override the values set for the BGP routing instance.
Specify the IP address of the interface used for BGP peering.	*update-source*	—
Configure IPv4 multicast or unicast address family attributes.	—	For the complete list of tasks used to configure IPv4 address family attributes, see Configuring IPv4 Address Family Attributes for a BGP Neighbor.
Configure IPv6 unicast address family attributes.	—	For the complete list of tasks used to configure IPv6 address family attributes, see Configuring IPv6 Address Family Attributes for a BGP Neighbor.
Configure the graceful restart characteristics.	—	For the complete list of tasks used to configure BGP graceful restart, see Configuring Graceful Restart Characteristics for a BGP Neighbor.
Configure the interval that must pass before the BGP routing process triggers fast-reset after all of the links in a BGP fast-reset interface list go down, and access BGP neighbor fast-reset configuration mode, where you can add up to ten links to a BGP fast-reset interface list.	*fast-reset*	BGP fast-reset occurs only when all interfaces in the BGP fast-reset interface list go down. If only one interface in a group goes down, an alternative path becomes active, and so on.
Add an interface to the BGP fast-reset interface list.	*interface*	Repeat this step to add additional interfaces to the BGP fast-reset interface list. You can add up to 10 interfaces to a list.

2.2.2 Configuring IPv4 Address Family Attributes for a BGP Neighbor

To configure the IPv4 address family attributes for a BGP neighbor, perform the tasks described in Table 8.

Note: Before you can configure IPv4 address family attributes for a BGP neighbor, you need to configure the BGP neighbor, as described in Configuring a BGP Neighbor.

Table 8 Configure IPv4 Address Family Attributes for a BGP Neighbor

Task	Root Command	Notes
Specify the use of standard IP Version 4 (IPv4) multicast or unicast address prefixes for the neighbors in the BGP address family, and to access BGP neighbor address family configuration mode.	**`address-family ipv4`** command See *address-family ipv4 (BGP)*.	Enter this command in BGP neighbor configuration mode.
Filter BGP routing updates from or to the specified BGP neighbor address family.	*as-path-list (BGP)*	—
Advertise the default route of the specified address family, even when the default route is not installed in the BGP routing table, to a BGP neighbor.	*default-originate*	—
Specify how the BGP routing process responds when the maximum number of prefixes sent by the BGP neighbor for the specified address family is exceeded.	*maximum prefix*	—

Table 8 Configure IPv4 Address Family Attributes for a BGP Neighbor

Task	Root Command	Notes
Apply the attributes of a configured BGP peer group to one or more BGP neighbor address families.	*peer-group*	A BGP neighbor address family can belong to more than one peer group and you can modify it to belong to a different peer group without having to delete the previous peer group association first. Attributes are inherited from the peer group to which a BGP neighbor address family is assigned. The following commands in BGP neighbor address family configuration mode represent attributes that you cannot customize per address family once it is assigned to a peer group: **`as-path-list out`**, **`prefix-list out`**, **`remove-private-as`**, and **`route-map out`**. Attributes inherited from a peer group that you can customize per neighbor address family include those set by the following commands: **`as-path-list in`**, **`default-originate`**, **`maximum-prefix, prefix-list in`**, and **`route-map in`**.
Filter BGP routes from or to the specified neighbor address family.	*prefix-list*	—
Remove ASNs from routes advertised to the specified BGP neighbor address family.	*remove-private-as*	—
Apply a route map that modifies BGP attributes or filters BGP routes received from or sent to the BGP neighbor.	*route-map (BGP)*	—
Configure an iBGP neighbor as a route reflector client for a BGP address family.	*route-reflector-clie nt*	—
Enable a BGP router to send MPLS labels with BGP IPv4 routes to a peer BGP router.	*send label*	The **`send label`** command is available for BGP routers configured with IPv4 unicast address prefixes only; it is not available for routers configured with multicast IPv4 address prefixes.

2.2.3 Configuring IPv6 Address Family Attributes for a BGP Neighbor

To configure the IPv6 address family attributes for a BGP neighbor, perform the tasks described in Table 9.

Note: Before you can configure IPv6 address family attributes for a BGP neighbor, you need to configure the BGP neighbor, as described in Configuring a BGP Neighbor.

Table 9 Configure IPv6 Address Family Attributes for a BGP Neighbor

Task	Root Command	Notes
Specify the use of standard IPv6 unicast address prefixes for the neighbors in the BGP address family, and to access BGP neighbor address family configuration mode.	*address-family ipv6 unicast*	Enter this command in BGP neighbor configuration mode.
Filter BGP routing updates from or to the specified BGP neighbor address family.	*as-path-list (BGP)*	—
Advertise the default route of the specified address family, even when the default route is not installed in the BGP routing table, to a BGP neighbor.	*default-originate*	—
Specify how the BGP routing process responds when the maximum number of prefixes sent by the BGP neighbor for the specified address family is exceeded.	*maximum prefix*	—

Table 9 Configure IPv6 Address Family Attributes for a BGP Neighbor

Task	Root Command	Notes
Apply the attributes of a configured BGP peer group to one or more BGP neighbor address families.	*peer-group*	A BGP neighbor address family can belong to more than one peer group and you can modify it to belong to a different peer group without having to delete the previous peer group association first. Attributes are inherited from the peer group to which a BGP neighbor address family is assigned. The following commands in BGP neighbor address family configuration mode represent attributes that you cannot customize per address family once it is assigned to a peer group: **`as-path-list out`**, **`prefix-list out`**,**`remove-private-as`**, and **`route-map out`**. Attributes inherited from a peer group that you can customize per neighbor address family include those set by the following commands: **`as-path-list in`**, **`default-originate`**, **`maximum-prefix`**, **`prefix-list in`**, and**`route-map in`**.
Filter BGP routes from or to the specified neighbor address family.	*prefix-list*	—
Remove ASNs from routes advertised to the specified BGP neighbor address family.	*remove-private-as*	—
Apply a route map that modifies BGP attributes or filters BGP routes received from or sent to the BGP neighbor.	*route-map (BGP)*	—

Table 9 Configure IPv6 Address Family Attributes for a BGP Neighbor

Task	Root Command	Notes
Configure an iBGP neighbor as a route reflector client for a BGP address family.	*route-reflector-client*	—
Enable a BGP router to send MPLS labels with BGP IPv6 routes to a peer BGP router.	*send label*	You must configure this command on both the local router and the peer router in order for the routers to send IPv6 unicast routes with MPLS labels. Before you use the **`send label`** command to enable the sending of IPv6 packets over an IPv4 core, you must enable MPLS on the core.

2.2.4 Configuring Graceful Restart Characteristics for a BGP Neighbor

Graceful restart is always enabled on all BGP routing instances, and is supported for both IPv4 and IPv6 address families. You cannot disable BGP graceful restart on a BGP neighbor, but you can configure certain characteristics.

To configure the graceful restart characteristics for a BGP neighbor, perform the tasks described in Table 10.

Note: Before you can configure graceful restart for a BGP neighbor, you need to configure the BGP neighbor, as described in Configuring a BGP Neighbor.

Table 10 Configure Graceful Restart Characteristics for a BGP Neighbor

Task	Root Command	Notes
Set the maximum amount of time after the local BGP speaker has been reset before it attempts to reconnect with the remote peer.	*maximum restart-time*	—

Table 10 Configure Graceful Restart Characteristics for a BGP Neighbor

Task	Root Command	Notes
Set the maximum amount of time the local BGP speaker retains routes it has previously received from a remote peer once that remote peer restarts the connection.	*maximum retain-time*	Any routes that have not been updated by the remote peer are deleted by the local peer after the local peer receives the end-of-RIB marker from the remote peer, or after the timer expires.
Force a BGP neighbor to retain routes from an iBGP peer once the peer has restarted.	*retain-ibgp-routes*	By default, routes are not retained for an iBGP peer after the peer restarts unless all iBGP peers support a graceful restart; however, in some network topologies, it may be desirable and feasible to retain the routes for an iBGP peer, even if not all iBGP peers support a graceful restart.

2.3 Enabling IPv6 over an IPv4 MPLS Core

To enable IPv6 over an IPv4 MPLS core, perform the tasks described in Table 11.

Note: IPv6 over IPv4 MPLS configuration is supported in the local context only.

Table 11 Enable IPv6 over an IPv4 Core

Task	Root Command	Notes
Access global configuration mode.	*configure*	Enter this command in global exec mode.
Enter context configuration mode.	*context* `local`	Enter this command in global configuration mode.
Enter router configuration mode for the specified BGP routing instance.	*router bgp*	Enter this command in context configuration mode.
Specify the use of IP Version 6 (IPv6) unicast address prefixes for the Border Gateway Protocol (BGP) routing instance and enter BGP address family configuration mode.	*address-family ipv6 unicast*	Enter this command in BGP router configuration mode.
Exit BGP address family configuration mode and enter BGP router configuration mode.	*exit*	Enter this command in BGP address family configuration mode.

Table 11 Enable IPv6 over an IPv4 Core

Task	Root Command	Notes
Enter BGP neighbor configuration mode for the specified neighbor.	*neighbor (BGP)*	Enter this command in BGP router configuration mode.
Specify the use of IPv6 unicast address prefixes for the specified BGP neighbor, and enter BGP neighbor address family configuration mode.	*address-family ipv6 unicast*	Enter this command in BGP neighbor configuration mode.
Enable the transport of labeled IPv6 routes over the MPLS IPv4 core.	*send label*	Enter this command in BGP neighbor address family configuration mode.

2.4 Configuring BGP Peer Groups and Peer Group Attributes

BGP peer groups are helpful in cases where many BGP neighbors are configured with the same update policies. Grouping a large number of neighbors into one or more peer groups simplifies modifications to a configuration and makes the BGP update calculation process more efficient. A BGP peer group can be an eBGP or as an iBGP peer group.

To configure a BGP peer group and other peer group attributes, perform the tasks described in the sections that follow.

2.4.1 Configuring a BGP Peer Group

To configure a BGP peer group, perform the tasks described in Table 12.

Table 12 Configure a BGP Peer Group

Task	Root Command	Notes
Configure a BGP peer group, and enter BGP peer group configuration mode.	*peer-group*	Enter this command in BGP router configuration mode.
Modify the minimal interval at which BGP routing updates are sent to the specified BGP peer group.	*advertisement-interval*	Range is from 0 to 600. For external BGP (eBGP), the default value is 30. For internal BGP (iBGP), the default value is 5.
Associate a description with the peer group.	*description (BGP)*	—
Configure the maximum number of hops used to reach an eBGP neighbor when the BGP peer group is not directly connected.	*ebgp-multihop*	This command must be enabled for BGP connections to be established with neighbors that are not directly connected.

Table 12 Configure a BGP Peer Group

Task	Root Command	Notes
Enable the BGP TTL security check in the kernel for the BGP peer group.	*enforce ttl*	For the BGP TTL security check to function correctly, it must be enabled on both ends of an eBGP session. Enabling only one end causes the eBGP session to drop.
Advertise the local peer address as the next-hop address.	*next-hop-self*	—
Configure an encrypted MD5 password for the BGP peer group.	*password (BGP)*	—
Send the community attribute to the specified BGP peer group.	*send community*	—
Enable a flapping peer to be temporarily suppressed for a configurable amount of time.	*session-dampening*	This command is per peer and peer-group based. If the peer is member of a peer group, the command is inherited from the peer-group and can be customized in the peer configuration. The main benefit of this feature is to avoid flapping peers from using system resources, and also to reduce routing churn induced by a flapping peer.
Administratively shut down a BGP session with the specified peer group.	*shutdown (BGP)*	This command temporarily shuts down a BGP session without removing a BGP peer group from the configuration.
Modify keepalive and holdtime timers for a peer group.	*timers keepalive*	—
Specifies the interval the router waits for a BGP peer to come up after a graceful restart before starting the best path computation.	*timers active-open*	Range is from 1 through 600 seconds. Be aware that the **`timers active-open`** interval configuration in BGP neighbor configuration mode takes precedence over the **`timers active-open`** interval configuration in BGP peer group configuration mode.

Table 12 Configure a BGP Peer Group

Task	Root Command	Notes
Specify the IP address of the interface used for BGP peering.	*update-source*	By default, when a BGP peer group receives BGP routes from an eBGP peer group, routes are sent to iBGP neighbors without changing the next-hop address.
Configure IPv4 multicast or unicast address family attributes.	For the complete list of tasks used to configure IPv4 address family attributes, see Configure IPv4 Address Family Attributes for a BGP Peer Group.	
Configure IPv6 unicast address family attributes.	For the complete list of tasks used to configure IPv6 address family attributes, see Configure IPv6 Address Family Attributes for a BGP Peer Group.	

2.4.2 Configuring IPv4 Address Family Attributes for a BGP Peer Group

To configure IPv4 address family attributes for a BGP peer group, perform the tasks described in Table 13. Enter all commands in BGP peer group address family configuration mode, unless otherwise noted.

Table 13 Configure IPv4 Address Family Attributes for a BGP Peer Group

Task	Root Command	Notes
Specify the use of standard IPv4 multicast or unicast address prefixes for peer groups in the BGP peer groups address family, and enter BGP peer group address family configuration mode.	**`address-family ipv4`** command See *address-family ipv4 (BGP)*.	Enter this command in BGP peer group configuration mode.
Filter BGP routing updates from or to the specified BGP neighbor address family.	*as-path-list (BGP)*	—
Advertise the default route of the specified address family, even when the default route is not installed in the BGP routing table, to a BGP neighbor.	*default-originate*	—
Specify how the BGP address family responds when the maximum number of prefixes sent by the BGP peer group for the specified address family is exceeded.	*maximum prefix*	—
Filter BGP routes from the peer group for the specified address family.	*prefix-list*	—

Table 13 Configure IPv4 Address Family Attributes for a BGP Peer Group

Task	Root Command	Notes
Remove ASNs from routes advertised to the specified BGP peer group address family.	*remove-private-as*	—
Apply a route map that modifies BGP attributes or filters BGP routes received from or sent to the specified peer group address family.	*route-map (BGP)*	—
Configure an iBGP peer group as a route reflector client for a BGP address family.	*route-reflector-client*	—

2.4.3 Configuring IPv6 Address Family Attributes for a BGP Peer Group

To configure IPv6 address family attributes for a BGP peer group, perform the tasks described in Table 14. Enter all commands in BGP peer group address family configuration mode, unless otherwise noted.

Table 14 Configure IPv6 Address Family Attributes for a BGP Peer Group

Task	Root Command	Notes
Specify the use of standard IPv6 unicast address prefixes for peer groups in the BGP peer groups address family, and enter BGP peer group address family configuration mode.	*address-family ipv6 unicast*	Enter this command in BGP peer group configuration mode.
Filter BGP routing updates from or to the specified BGP neighbor address family.	*as-path-list (BGP)*	—
Advertise the default route of the specified address family, even when the default route is not installed in the BGP routing table, to a BGP neighbor.	*default-originate*	—
Specify how the BGP address family responds when the maximum number of prefixes sent by the BGP peer group for the specified address family is exceeded.	*maximum prefix*	—
Filter BGP routes from the peer group for the specified address family.	*prefix-list*	—

Table 14 Configure IPv6 Address Family Attributes for a BGP Peer Group

Task	Root Command	Notes
Remove ASNs from routes advertised to the specified BGP peer group address family.	*remove-private-as*	—
Apply a route map that modifies BGP attributes or filters BGP routes received from or sent to the specified peer group address family.	*route-map (BGP)*	—
Configure an iBGP peer group as a route reflector client for a BGP address family.	*route-reflector-client*	—

2.4.4 Applying Peer Group Attributes

A BGP neighbor, or BGP neighbor address family, can inherit attributes from the peer group to which a neighbor is assigned. The following BGP neighbor configuration mode commands represent attributes that cannot be customized per neighbor when the neighbor is assigned to a peer group: **`advertisement-interval`**, **`ebgp-multihop`**, **`local-as`**, **`send community`**, and **`timers keepalive`**. Attributes inherited from a peer group that can be customized per neighbor include those set by the following commands: **`description`**, **`password`**, **`send prefix`**, **`shutdown`**, and **`update-source`**.

To apply peer group attributes, perform the tasks described in Table 15.

Table 15 Apply Peer Group Attributes

Task	Root Command	Notes
Apply peer group attributes to a BGP neighbor.	*peer-group*	Enter this command in BGP neighbor configuration mode.
Apply peer group attributes to a BGP neighbor address family.	*peer-group*	Enter this command in BGP peer group configuration mode.

2.5 Configuring BGP Prefix-Based ORF

To enable BGP prefix-based ORF, perform the tasks described in Table 16.

Table 16 Configuring BGP ORF

#	Task	Root Command	Notes
Configure an IP-prefix list with ORF filters:			

Table 16 Configuring BGP ORF

#	Task	Root Command	Notes
1.	Access global configuration mode.	*configure*	Enter this command in global exec mode.
2.	Enter context configuration mode.	*context*	Enter this command in global configuration mode.
3.	Create an IP prefix list and access IP prefix list configuration mode.	*ip prefix-list*	When the prefix list is applied to a BGP neighbor, that neighbor takes on the filters specified by the IP prefix list.
4.	Create a filter.	seq	The sequence number determines the order in which the filter is applied. You can configure a filter that permits or denies BGP updates from a specified IP whose prefix length is equal to, greater than, or less than the specified prefix length.
5.	Repeat Step 1 through Step 4 to add more filters to an IP prefix list.		
Apply the prefix list to a BGP neighbor:			
6.	Access global configuration mode.	*configure*	Enter this command in global exec mode.
7.	Enter context configuration mode.	*context*	Enter this command in global configuration mode.
8.	Access router configuration mode for the specified BGP routing instance.	*router bgp*	—
9.	Enter BGP neighbor configuration mode for the specified neighbor.	*neighbor (BGP)*	—
10.	Advertise to a BGP peer that this BGP speaker can send prefixed-based filtering to the peer.	*send filter prefix-list*	Because ORF capabilities are communicated between BGP speakers during BGP connection establishment, the *send filter prefix-list* command does not take effect until the BGP connection is reset.

Table 16 Configuring BGP ORF

#	Task	Root Command	Notes
11.	Access BGP neighbor address family configuration mode.	**address-family ipv4** command (See *address-famil y ipv4 (BGP)* or See *address-family ipv6 unicast* command. or See *address-family ipv6 vpn* command.	
12.	Apply the IP prefix list you configured in Steps 1 through 5 to the neighbor address family.	*prefix-list* ***pl-name*** **in**	
Configure the receiving BGP peer (another BGP speaker) to accept ORFs received from the sending BGP speaker:			
13.	Access global configuration mode.	*configure*	Enter this command in global exec mode.
14.	Enter context configuration mode.	*context*	Enter this command in global configuration mode.
15.	Access router configuration mode for a BGP routing instance.	*router bgp*	—
16.	Advertise to a peer that this BGP speaker is willing to accept address prefix-based route filtering from the peer.	*accept filter prefix-list*	Because ORF capabilities are communicated between BGP speakers during BGP connection establishment, the *accept filter prefix-list* command does not take effect until the BGP connection is reset.

2.6 BGP Operations

To manage BGP functions, perform the appropriate tasks described in Table 17. Enter the **show** commands in any mode; enter the **clear** and **debug** commands in exec mode.

Table 17 BGP Operations Tasks

Task	Root Command
Apply new BGP routing policies or reset BGP connections globally without dropping connections.	*clear bgp*
Apply new routing policies for eBGP neighbors or to reset eBGP connections.	*clear bgp external*
Clear BGP route-flap statistics.	*clear bgp flap-statistics*
Apply new BGP routing policies to connections using multicast address prefixes or to reset BGP IPv4 address connections.	*clear bgp ipv4 multicast*
Apply new BGP routing policies to connections using unicast address prefixes or to reset BGP IPv4 address connections.	*clear bgp ipv4 unicast*
Clear BGP message statistics.	*clear bgp ipv4 vpn*
Apply new BGP neighbor routing policies or reset BGP neighbor connections.	*clear bgp neighbor*
Apply new BGP peer group routing policies or to reset BGP peer group connections.	*clear bgp peer-group*
Enable the generation of BGP general-event messages.	*debug bgp event*
Enable the generation of debug messages for BGP passive open connections.	*debug bgp listen*
Enable the generation of debug messages for BGP nonupdate events.	*debug bgp message*
Enable the generation of debug messages for BGP routing policies.	*debug bgp policy*
Enable the generation of debug messages for interaction between BGP and the Routing Information Base (RIB).	*debug bgp rib*
Enable the generation of debug messages for BGP session states and timers.	*debug bgp session-state*
Enable the generation of debug messages for BGP update events.	*debug bgp update*
Display BGP attribute information, including AS path, community, next-hop address, and route reflector attributes.	*show bgp attribute*
Display malformed BGP messages for the purpose of troubleshooting.	*show bgp malform*
Display BGP neighbor status, configuration, and statistical information.	*show bgp neighbor*
Display BGP neighbor flap statistics.	*show bgp neighbor flap-statistics*
Display BGP notification messages.	*show bgp notification*
Display BGP peer group information, including peer group membership and session status.	*show bgp peer-group*

Table 17 BGP Operations Tasks

Task	Root Command
Display BGP neighbor reset information for troubleshooting purposes.	*show bgp reset-log*
Display information about all BGP routes, or for a subset of routes.	*show bgp route*
Display community information for BGP routes.	*show bgp route community*
Display BGP route-flap statistics.	*show bgp route flap-statistics*
Display BGP routes sourced from more than one AS.	*show bgp route inconsistent-as*
Display information about BGP multicast or unicast IP Version 4 (IPv4) address prefix-based routes.	*show bgp route ipv4*
Display information about BGP unicast IP Version 6 (IPv6) address prefix-based routes.	*show bgp route ipv6 unicast*
Display MPLS labels associated with BGP routes.	*show bgp route labels*
Display information about routes to or from BGP neighbors.	*show bgp route neighbor*
Display BGP communities that match an AS path string.	*show bgp route regexp*
Display BGP routes sourced from the local AS.	*show bgp route sourced*
Display a summary report of BGP routes in the routing table.	*show bgp route summary*
Display a summary of BGP status and statistical information.	*show bgp summary*
Display the current BGP configuration information for the current context.	*show configuration bgp*

Caution!

Risk of dropped connection. A hard reset can impact network connectivity. When using any **`clear bgp`** command, the **`soft`** keyword for inbound only takes effect if the BGP neighbor supports the refresh capability. The **`soft`** keyword for outbound is a local matter, and does not require the capability. To see if a BGP neighbor supports the refresh capability, use the **`show bgp neighbor summary`** command (in exec mode). Specify the **`soft`** keyword if you do not want the BGP neighbor connection dropped. To reduce the risk, only use a hard reset as a last resort.

3 Configuration Examples

The sections that follow provide BGP configuration examples for basic BGP, next-hop triggered BGP best-path calculation, iMP-BGP peers, iMP-BGP peer groups, eMP-BGP peers, eMP-BGP peer groups, and IPv6 over an IPv4 core.

3.1 Example: Configure Basic BGP

The following example show the minimum commands needed to configure BGP:

```
[local]Router_A#config
[local]Router_A(config)#context local
[local]Router_A(config-ctx)#router bgp 64001
[local]Router_A(config-bgp)#router-id 1.1.1.71
[local]Router_A(config-bgp)#address-family ipv4 unicast
[local]Router_A(config-bgp-af)#redistribute static
[local]Router_A(config-bgp-af)#exit
[local]Router_A(config-bgp)#peer-group iBGP internal
[local]Router_A(config-bgp-peer-group)#next-hop-self
[local]Redback(config-bgp-peer-group)#update-source loopback0
[local]Redback(config-bgp-peer-group)#address-family ipv4 unicast
[local]Redback(config-bgp-peer-af)#exit
[local]Redback(config-bgp-peer-group)#exit
[local]Redback(config-bgp)#peer-group customer-routes external
[local]Redback(config-bgp-peer-group)#address-family ipv4 unicast
[local]Redback(config-bgp-peer-af)#route-map rmap1 out
[local]Redback(config-bgp-peer-af)#exit
[local]Redback(config-bgp-peer-group)#exit
```

```
[local]Redback(config-bgp)#neighbor 1.1.1.1 internal
[local]Redback(config-bgp-neighbor)#peer-group ibgp
[local]Redback(config-bgp-neighbor)#exit
[local]Redback(config-bgp)#neighbor 2.2.2.2 external
[local]Redback(config-bgp-neighbor)#remote-as 200
[local]Redback(config-bgp-neighbor)#peer-group customer-routes
[local]Redback(config-bgp-neighbor)#address-family ipv4 unicast
[local]Redback(config-bgp-peer-af)#prefix-list bar in
[local]Redback(config-bgp-peer-af)#route-map foo2 in
[local]Redback(config-bgp-peer-af)#exit
[local]Redback(config-bgp-neighbor)#exit
[local]Redback(config-bgp)#neighbor 3.3.3.3 external
[local]Redback(config-bgp-neighbor)#remote-as 300
[local]Redback(config-bgp-neighbor)#address-family ipv4 unicast
[local]Redback(config-bgp-peer-af)#prefix-list bar in
[local]Redback(config-bgp-peer-af)#route-map foo3 out
```

3.2 Example: Configure Next-Hop-Triggered BGP Best-Path Calculation

The following example shows how to enable and then configure next-hop triggered BGP best-path calculation:

```
[local]Router_A#config
[local]Router_A(config)#context local
[local]Router_A(config-ctx)#router bgp 64001
[local]Router_A(config-bgp)#address-family ipv4 unicast
[local]Router_A(config-bgp-af)#nexthop triggered
[local]Router_A(config-bgp-af)#nexthop triggered delay 30
[local]Router_A(config-bgp-af)#nexthop triggered holdtime 2 backoff 10
[local]Router_A(config-bgp-af)#commit
```

3.3 Example: Configure iMP-BGP Peers

The following example configures two iMP-BGP peers. Figure 6 shows the network topology for the configuration.

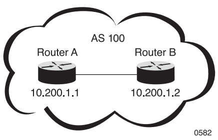

Figure 6 Network Topology for iMP-BGP Peer Configuration

The configuration for **Router_A** is as follows:

```
[local]Router_A#config
[local]Router_A(config)#context local
[local]Router_A(config-ctx)#interface lo1 loopback
[local]Router_A(config-if)#ip address 10.200.1.1/32
[local]Router_A(config-if)#exit
[local]Router_A(config-ctx)#router bgp 100
[local]Router_A(config-bgp)#router-id 10.200.1.1
[local]Router_A(config-bgp)#neighbor 10.200.1.2 internal
[local]Router_A(config-bgp-neighbor)#update-source lo1
[local]Router_A(config-bgp-neighbor)#address-family ipv4 multicast
[local]Router_A(config-bgp-peer-af)#exit
[local]Router_A(config-bgp-neighbor)#exit
[local]Router_A(config-bgp)#exit
[local]Router_A(config-ctx)#ip route 10.200.1.2/32 102.1.1.2
```

The configuration for **Router B** is as follows:

```
[local]Router_B#config
[local]Router_B(config)#context local
[local]Router_B(config-ctx)#interface lo1 loopback
[local]Router_B(config-if)#ip address 10.200.1.2/32
[local]Router_B(config-if)#exit
[local]Router_B(config-ctx)#router bgp 100
[local]Router_B(config-bgp)#router-id 10.200.1.2
[local]Router_B(config-bgp)#neighbor 10.200.1.1 internal
[local]Router_B(config-bgp-neighbor)#update-source lo1
[local]Router_B(config-bgp-neighbor)#address-family ipv4 multicast
[local]Router_B(config-bgp-peer-af)#exit
[local]Router_B(config-bgp-neighbor)#exit
[local]Router_B(config-bgp)#exit
[local]Router_B(config-ctx)#ip route 10.200.1.1/32 102.1.1.1
```

3.4 Example: Configure an iMP-BGP Peer Group

The following example configures an iMP-BGP peer group for two iMP-BGP peers. Figure 7 shows the network topology for the configuration.

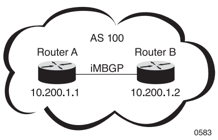

Figure 7 Network Topology for iMP-BGP Peer Group Configuration

The configuration for **Router_A** is as follows:

```
[local]Router_A#config
[local]Router_A(config)#context local
[local]Router_A(config-ctx)#interface lo1 loopback
[local]Router_A(config-if)#ip address 10.200.1.1/32
[local]Router_A(config-if)#exit
[local]Router_A(config-ctx)#router bgp 100
[local]Router_A(config-bgp)#router-id 10.200.1.1
[local]Router_A(config-bgp)#address-family ipv4 multicast
[local]Router_A(config-bgp-af)#exit
[local]Router_A(config-bgp)#peer-group iMBGP internal
[local]Router_A(config-bgp-peer-group)#update-source lo1
[local]Router_A(config-bgp-peer-group)#address-family ipv4 multicast
[local]Router_A(config-bgp-peer-af)#exit
[local]Router_B(config-bgp-peer-group)#exit
[local]Router_A(config-bgp)#neighbor 10.200.1.2 internal
[local]Router_A(config-bgp-neighbor)#peer-group iMBGP
```

The configuration for **Router_B** is as follows:

```
[local]Router_B#config
[local]Router_B(config)#context local
[local]Router_B(config-ctx)#interface lo1 loopback
[local]Router_B(config-if)#ip address 10.200.1.2/32
[local]Router_B(config-if)#exit
[local]Router_B(config-ctx)#router bgp 100
[local]Router_B(config-bgp)#router-id 10.200.1.2
[local]Router_B(config-bgp)#address-family ipv4 multicast
[local]Router_B(config-bgp-af)#exit
[local]Router_B(config-bgp)#peer-group iMBGP internal
[local]Router_B(config-bgp-peer-group)#update-source lo1
[local]Router_B(config-bgp-peer-group)#address-family ipv4 multicast
[local]Router_B(config-bgp-peer-af)#exit
[local]Router_B(config-bgp-peer-group)#exit
[local]Router_B(config-bgp)#neighbor 10.200.1.1 internal
[local]Router_B(config-bgp-neighbor)#peer-group iMBGP
```

3.5 Example: Configure eMP-BGP Peers

The following example configures two eMP-BGP peers. Figure 8 shows the network topology for the configuration.

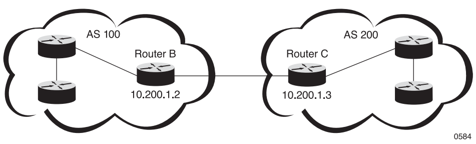

Figure 8 Network Topology for Configuring eMP-BGP Peers

The configuration for **Router_B** is as follows:

```
[local]Router_B#config
[local]Router_B(config)#context local
[local]Router_B(config-ctx)#interface lo1 loopback
[local]Router_B(config-if)#ip address 10.200.1.2/32
[local]Router_B(config-if)#exit
[local]Router_B(config-ctx)#router bgp 100
[local]Router_B(config-bgp)#router-id 10.200.1.2
[local]Router_B(config-bgp)#neighbor 10.200.1.3 external
[local]Router_B(config-bgp-neighbor)#remote-as 200
[local]Router_B(config-bgp-neighbor)#ebgp-multihop 10
[local]Router_B(config-bgp-neighbor)#update-source lo1
[local]Router_B(config-bgp-neighbor)#address-family ipv4 multicast
```

The configuration for **Router_C** is as follows:

```
[local]Router_C#config
[local]Router_C(config)#context local
[local]Router_C(config-ctx)#interface lo1 loopback
[local]Router_C(config-if)#ip address 10.200.1.3/32
[local]Router_C(config-if)#exit
[local]Router_C(config-ctx)#router bgp 100
[local]Router_C(config-bgp)#router-id 10.200.1.2
[local]Router_C(config-bgp)#neighbor 10.200.1.1 internal
[local]Router_C(config-bgp-neighbor)#remote-as 100
[local]Router_C(config-bgp-neighbor)#ebgp-multihop 10
[local]Router_C(config-bgp-neighbor)#update-source lo1
[local]Router_C(config-bgp-neighbor)#address-family ipv4 multicast
```

3.6 Example: Configure an eMP-BGP Peer Group

The following example configures an eMP-BGP peer group for two eMP-BGP peers. Figure 9 shows the network topology for the configuration.

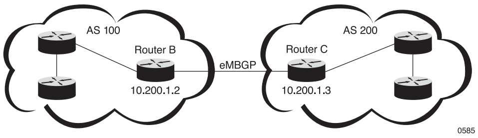

Figure 9 Network Topology for Configuring an eMP-BGP Peer Group

The configuration for **Router_B** is as follows:

```
[local]Router_B#config
[local]Router_B(config)#context local
[local]Router_B(config-ctx)#interface lo1 loopback
[local]Router_B(config-if)#ip address 10.200.1.2/32
[local]Router_B(config-if)#exit
[local]Router_B(config-ctx)#router bgp 100
[local]Router_B(config-bgp)#router-id 10.200.1.2
[local]Router_B(config-bgp)#address-family ipv4 multicast
[local]Router_B(config-bgp-af)#exit
[local]Router_B(config-bgp)#peer-group eMBGP external
[local]Router_B(config-bgp-peer-group)#ebgp-multihop 10
[local]Router_B(config-bgp-peer-group)#update-source lo1
[local]Router_B(config-bgp-peer-group)#address-family ipv4 multicast
[local]Router_B(config-bgp-peer-af)#exit
[local]Router_B(config-bgp-peer-group)#neighbor 10.200.1.3 external
[local]Router_B(config-bgp-neighbor)#remote-as 200
[local]Router_B(config-bgp-neighbor)#peer-group eMBGP
```

The configuration for **Router_C** is as follows:

```
[local]Router_C#config
[local]Router_C(config)#context local
[local]Router_C(config-ctx)#interface lo1 loopback
[local]Router_C(config-if)#ip address 10.200.1.3/32
[local]Router_C(config-if)#exit
[local]Router_C(config-ctx)#router bgp 200
[local]Router_C(config-bgp)#router-id 10.200.1.3
[local]Router_C(config-bgp)#address-family ipv4 multicast
[local]Router_C(config-bgp-af)#exit
[local]Router_C(config-bgp)#peer-group eMBGP external
[local]Router_C(config-bgp-peer-group)#ebgp-multihop 10
[local]Router_C(config-bgp-peer-group)#update-source lo1
[local]Router_C(config-bgp-peer-group)#address-family ipv4 multicast
[local]Router_C(config-bgp-peer-af)#exit
[local]Router_C(config-bgp-peer-group)#neighbor 10.200.1.2 external
[local]Router_C(config-bgp-neighbor)#remote-as 100
[local]Router_C(config-bgp-neighbor)#peer-group eMBGP
```

3.7 Example: Configure IPv6 over an IPv4 Core

The following example shows how to enable IPv6 over an MPLS IPv4 core in the SmartEdge router. Perform this configuration on a PE router that has at least one IPv4 session with a PE neighbor and one IPv6 session with a CE neighbor:

```
[local]Router_C#config context local

[local]Router_A(config-ctx)#router bgp 100

[local]Router_C(config-bgp)#address-family ipv6 unicast

[local]Router_C(config-bgp-af)#exit

[local]Router_C(config-bgp)#neighbor 10.200.1.3 internal

[local]Router_C(config-bgp-neighbor)#address-family ipv6 unicast

[local]Router_C(config-bgp-peer-af)#send label

[local]Router_C(config-bgp-peer-af)#commit
```

3.8 Example: Configure BGP ORF

The following example shows how to configure BGP ORF between a BGP speaker (IP address `192.168.255.120`) and its BGP peer (IP address `192.168.255.110`):

The following example creates an IP prefix list called **`ORF-test`** that has three filters:

```
[local]Redback#configure
[local]Redback(config)#context local
[local]Redback(config-ctx)#ip prefix-list ORF-test
[local]Redback(config-prefix-list)#seq 10 permit 192.168.128.0/24 ge 24
[local]Redback(config-prefix-list)#seq 15 permit 192.168.127.0/24 le 18
[local]Redback(config-prefix-list)#seq 20 deny any
```

The following example advertises to a BGP neighbor that this BGP speaker can send prefixed-based filtering to the peer, and applies the IP prefix list **`ORF-test`** to the BGP peer:

```
[local]Redback#configure
[local]Redback(config)#context local
[local]Redback(config-ctx)#router bgp 999
[local]Redback(config-bgp)#neighbor 192.168.255.110 external
[local]Redback(config-bgp-neighbor)#address-family ipv4 unicast
[local]Redback(config-bgp-peer-af)#prefix-list ORF-test in
```

The following example configures the BGP peer (whose IP address is 192.168.255.110) to accept ORFs received from the sending BGP speaker (whose IP address is 192.168.255.120):

```
[local]Redback#configure
[local]Redback(config)#context local
[local]Redback(config-ctx)#router bgp 100
[local]Redback(config-bgp)#neighbor 192.168.255.120 external
[local]Redback(config-bgp-neighbor)#accept filter prefix-list
```

3.9 Example: Configure BGP Route Redistribution and Aggregation

The following example configures route redistribution and aggregation for a BGP routing instance. First, configure a list of aggregate IP prefixes:

```
[local]Router(config-ctx)#ipv6 prefix-list test1-aggregate
[local]Router(config-ipv6-prefix-list)#seq 10 permit 4001:101:101:106::/64 ge 64
[local]Router(config-ipv6-prefix-list)#seq 20 permit 5001:101:101:106::/64 ge 64
[local]Router(config-ipv6-prefix-list)#seq 30 permit 6001:101:101:106::/64 ge 64
[local]Router(config-ipv6-prefix-list)#seq 40 permit 7001:101:101:106::/64 ge 64
[local]Router(config-ipv6-prefix-list)#seq 50 permit 2001:101:101::/48 ge 48
```

Next, configure route map **test1** that aggregates the IPv6 prefixes in the aggregate prefix list called **test1-aggregate**:

```
[local]Router(config-ctx)#route-map test1 permit 10
[local]Router(config-route-map)#match ipv6 address prefix-list test1-aggregate
[local]Router(config-route-map)#set ipv6 aggregate test1-aggregate
```

Specify that routes selected for redistribution are summarized only if they contain any of the prefixes specified in IPv6 prefix list **test1**:

```
[local]Redback(config-ctx)#router bgp 1
[local]Redback(config-bgp)#address-family ipv6 unicast
[local]Redback(config-bgp-af)#redistribute static route-map test1
```

Configure the static routes. In this example, the routes match aggregate prefix **2001:101:101::/48**:

```
[local]Redback(config-ctx)#ipv6 route 2001:101:101:303::/64 80::2
[local]Redback(config-ctx)#ipv6 route 2001:101:101:304::/64 80::2
[local]Redback(config-ctx)#ipv6 route 2001:101:101:305::/64 80::2
[local]Redback(config-ctx)#ipv6 route 2001:101:101:306::/64 80::2
[local]Redback(config-ctx)#ipv6 route 2001:101:101:307::/64 80::2
```

Note: When an IP prefix list is used for aggregation, the **`ge`** and **`le`** parameters (configured with the **`seq`** command) are ignored, and the prefix list entries match any route subsumed by the prefix. In such cases, the **`ge`** parameter is implicit.