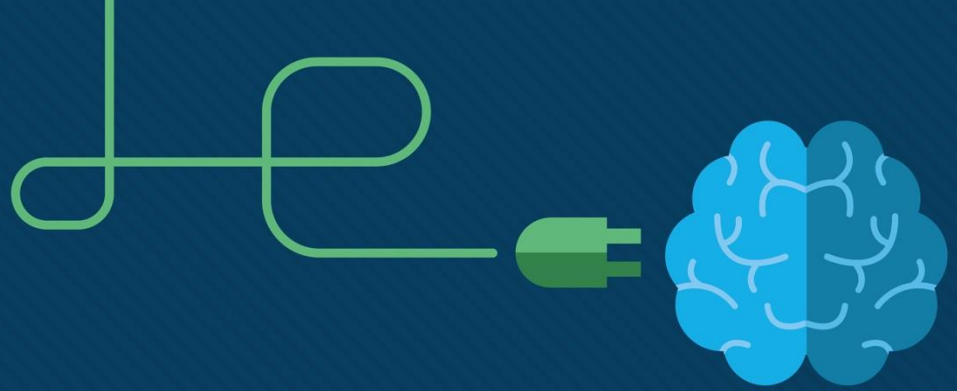

Chapter 11: BGP

Instructor Materials

CCNP Enterprise: Advanced Routing

Chapter 11 Content

This chapter covers the following content:

- **BGP Fundamentals -** This section provides an overview of the fundamentals of the BGP routing protocol.
- **Basic BGP Configuration -** This section walks through the process of configuring BGP to establish a neighbor session and how routes are exchanged between peers
- **Understanding BGP Session Types and Behaviors -** This section provides an overview of how route summarization works with BGP and some of the design considerations related to summarization.
- **Multiprotocol BGP for IPv6 -** This section explains how BGP provides support for IPv6 routing and its configuration.

BGP Fundamentals

- This chapter explains the core concepts of BGP and the basics of advertising routes with other organizations by using BGP.

Autonomous System Numbers (ASNs)

In BGP, an autonomous system (AS) is a collection of routers under a single organization's control. BGP can use one or more Interior Gateway Protocols (IGPs) and common metrics to route packets within an AS, although an AS can use BGP as the only routing protocol.

An organization requiring connectivity to the internet must obtain an ASN. ASNs were originally 2 bytes (in the 16-bit range) which made 65,535 ASNs possible. Due to exhaustion, RFC 4893 expanded the ASN field to accommodate 4 bytes (in the 32-bit range). This allows for 4,294,967,295 unique ASNs.

Two blocks of private ASNs are available for any organization to use, as long as these ASNs are never exchanged publicly on the internet.

- ASNs 64,512 through 65,535 are private ASNs in the 16-bit range
- ASNs 4,200,000,000 through 4,294,967,294 are private ASNs in the 32-bit range.

It is imperative to use only the ASN assigned by the IANA, the ASN assigned by your service provider, or a private ASN. Using another organization's ASN without permission could result in traffic loss and cause havoc on the internet.

BGP Sessions

A BGP session is an established adjacency between two BGP routers. Multi-hop sessions require that the router use an underlying route installed in the Routing Information Base (RIB) (static or from any routing protocol) to establish the TCP session with the remote endpoint.

BGP sessions are categorized into two types:

- Internal BGP (iBGP) - Sessions established with an iBGP router that are in the same AS or that participate in the same BGP confederation
- External BGP (eBGP) - Sessions established with a BGP router that is in a different AS

Path Attributes

BGP uses path attributes (PAs) associated with each network path. The PAs provide BGP with granularity and control of routing policies in BGP. The BGP prefix PAs are classified as any of the following:

- Well-known mandatory: must be included with every prefix advertisement
- Well-known discretionary: may or may not be included with the prefix advertisement
- Optional transitive: stays with the route advertisement from AS to AS
- Optional non-transitive: cannot be shared from AS to AS

Well-known attributes must be recognized by all BGP implementations. Optional attributes do not have to be recognized by all BGP implementations.

In BGP, the Network Layer Reachability Information (NLRI) is the routing update that consists of the network prefix, prefix length, and any BGP PAs for that specific route.

Loop Prevention

- BGP is a path vector routing protocol and does not contain a complete topology of the network, as do link-state routing protocols.
- BGP behaves like distance vector protocols, ensuring that a path is loop free.
- The BGP attribute AS_Path is a well-known mandatory attribute and includes a complete list of all the ASNs that the prefix advertisement has traversed from its source AS.
- AS_Path is used as a loop-prevention mechanism in BGP. If a BGP router receives a prefix advertisement with its AS listed in AS_Path, it discards the prefix because the router thinks the advertisement forms a loop.

Address Families

- BGP was intended for only routing IPv4 prefixes, but RFC 2858 added Multi-Protocol BGP (MP-BGP) capability by adding an extension called the address family identifier (AFI).
- An address family correlates to IPv4 or IPv6, and additional granularity is provided through a subsequent address family identifier (SAFI), such as unicast or multicast.
- Multiprotocol BGP (MP-BGP) achieves this separation by using the BGP path attributes (PAs) MP_REACH_NLRI and MP_UNREACH_NLRI. These attributes are held inside BGP update messages and are used to carry network reachability information for different address families.
- Every address family maintains a separate database and configuration for each protocol (address family plus sub-address family) in BGP. This allows for a routing policy in one address family to be different from a routing policy in a different address family.
- BGP includes an AFI and a SAFI with every route advertisement to differentiate between the AFI and SAFI databases.

Inter-Router Communication

- BGP does not use hello packets to discover neighbors, and it cannot discover neighbors dynamically. Inter-autonomous system routing adjacencies should not change frequently and are coordinated. A BGP session refers to the established adjacency between two BGP routers.
- BGP neighbors are defined by IP address and BGP uses TCP port 179 to communicate with other routers which can cross networks (multi-hop capable). BGP can form directly connected neighbor adjacencies as well as adjacencies that are multiple hops away.
- Multi-hop sessions require that the router use an underlying route installed in the RIB (static or from any routing protocol) to establish the TCP session with the remote endpoint.

In Figure 11-1, R1 is able to establish a direct BGP session with R2. In addition, R2 is able to establish a BGP session with R4, even though it passes through R3. R3 is unaware that R2 and R4 have established a BGP session even though the packets flow through R3.

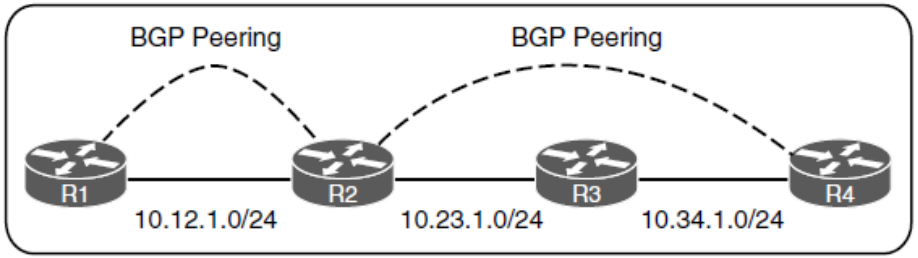

Figure 11-1 *BGP Single- and Multi-Hop Sessions*

BGP Messages

BGP communication uses four message types: OPEN, UPDATE, NOTIFICATION and KEEPALIVE as shown in Table 11-2.

Table 11-2 BGP Packet Types

Type	Name	Functional Overview
1	OPEN	Sets up and establishes BGP adjacency
2	UPDATE	Advertises, updates, or withdraws routes
3	NOTIFICATION	Indicates an error condition to a BGP neighbor
4	KEEPALIVE	Ensures that BGP neighbors are still alive

OPEN – OPEN is used to set up and establish a BGP adjacency. The OPEN message contains the BGP version number, the ASN of the originating router, the hold time, the BGP identifier, and other optional parameters that establish the session capabilities:

- **Hold time -** When establishing a BGP session, the routers use the smaller hold time value contained in the two routers' OPEN messages. The hold time value must be at least 3 seconds, or the hold time is set to 0 to disable KEEPALIVE messages. For Cisco routers, the default hold time is 180 seconds. When the hold timer reaches zero, the BGP session is torn down and routes are removed. An update message is sent to other BGP neighbors for the affected prefixes.

- **BGP identifier -** the BGP router ID (RID) is a 32-bit unique number that identifies the BGP router in the advertised prefixes. The RID can be used as a loop-prevention mechanism for routers advertised within an autonomous system. The RID can be set manually or dynamically for BGP. A nonzero value must be set in order for routers to become neighbors.

BGP Messages (Cont.)

KEEPALIVE - These messages are exchanged ever one-third of the hold timer agreed upon between the two BGP routers. Cisco devices have a default hold time of 180 seconds, and a default keepalive interval of 60 seconds. If the hold time is set to 0, no KEEPALIVE messages are sent between the BGP neighbors.

Table 11-2 BGP Packet Types

Type	Name	Functional Overview
1	OPEN	Sets up and establishes BGP adjacency
2	UPDATE	Advertises, updates, or withdraws routes
3	NOTIFICATION	Indicates an error condition to a BGP neighbor
4	KEEPALIVE	Ensures that BGP neighbors are still alive

BGP communication uses four message types, as shown in Table 11-2.

UPDATE – This message advertises any feasible routes, withdraws previously advertised routes, or both. The UPDATE message holds the NLRI, which includes the prefix and associated BGP PAs when advertising prefixes. Withdrawn NLRI routes include only the prefix. An UPDATE message can act as a keepalive to reduce unnecessary traffic.

NOTIFICATION – This message is sent when an error is detected with the BGP session, such as a hold timer expiring, neighbor capabilities changing, or a BGP session reset being requested. This causes the BGP connection to close.

BGP Neighbor States

BGP forms a TCP session with neighbor routers called peers. BGP uses the finite-state machine (FSM) shown in Figure 11-2, to maintain a table of all BGP peers and their operational status. The BGP session may report the following states:

Idle - BGP detects a start event and tries to initiate a TCP connection to the BGP peer and also listens for a new connection from a peer router. If an error causes BGP to go back to the Idle state a second time, the ConnectRetry timer is set to 60 seconds and must decrement to 0 before the connection can be initiated again. Further failures to leave the Idle state result in the ConnectRetry timer doubling in length from the previous time.

Connect - BGP initiates the TCP connection. The router initiating the request uses a dynamic source port, but the destination port is always 179. If the three-way TCP handshake completes, the established BGP session resets the ConnectRetry timer, sends the OPEN message to the neighbor, changes to the OpenSent state. During the connect state, the router with the higher IP address manages the connection

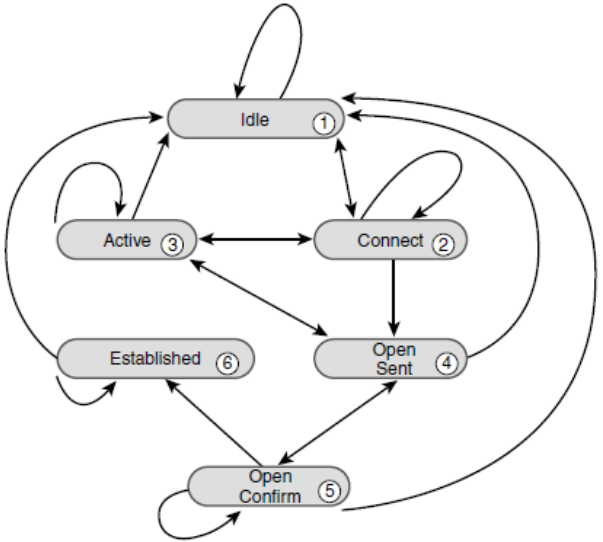

Figure 11-2 *BGP Finite-State Machine*

BGP Neighbor States (Cont.)

Active - BGP starts a new three-way TCP handshake. If a connection is established, an OPEN message is sent, the hold timer is set to 4 minutes, and the state moves to OpenSent. If this attempt for TCP connection fails, the state moves back to the Connect state and resets the ConnectRetry timer.

OpenSent - An OPEN message has been sent and is awaiting an OPEN message from the other router. When the router receives the OPEN message from the other router, both OPEN messages are checked for errors. The following items are compared:

- BGP versions must match.
- OPEN message source IP must match the configured neighbor IP.
- OPEN message AS number must match the configured neighbor AS.
- BGP identifiers (RIDs) must be unique. If no RID exists condition is not met.
- Security parameters (such as password and TTL) must be set correctly.

If the OPEN messages do not have any errors, the hold time is negotiated and a KEEPALIVE message is sent. The connection state is then moved to OpenConfirm. If an error is found in the OPEN message, a NOTIFICATION message is sent, and the state is moved back to Idle. If TCP receives a disconnect message, BGP closes the connection, resets the ConnectRetry timer, and sets the state to Active.

BGP Neighbor States (Cont.)

OpenConfirm - BGP waits for a KEEPALIVE or NOTIFICATION message. Upon receipt of a neighbor's KEEPALIVE, the state is moved to Established. If the hold timer expires, a stop event occurs, or a NOTIFICATION message is received, the state is moved to Idle.

Established - The BGP session is established. BGP neighbors exchange routes through UPDATE messages. As UPDATE and KEEPALIVE messages are received, the hold timer is reset. If the hold timer expires, an error is detected, and BGP moves the neighbor back to the Idle state.

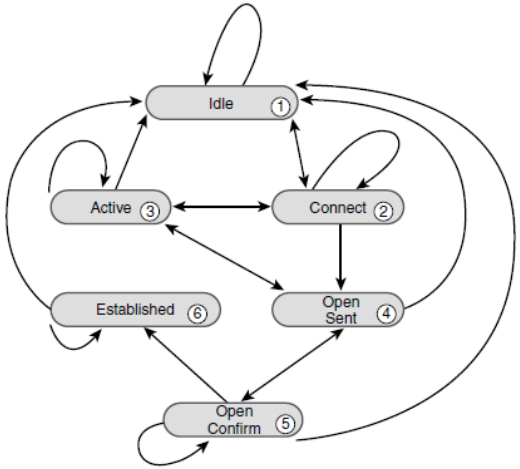

Figure 11-2 *BGP Finite-State Machine*

Example 11-1 shows an established BGP session with the command **show tcp brief** displaying the active TCP sessions between a router. Notice that the TCP source port is 179, and the destination port is 59884 on R1; the ports are opposite on R2.

Example 11-1 *Established BGP Session*

```
R1# show tcp brief
TCB        Local Address         Foreign Address       (state)
F6F84258   10.12.1.1.59884       10.12.1.2.179         ESTAB

R2# show tcp brief
TCB        Local Address         Foreign Address       (state)
EF153B88   10.12.1.2.179         10.12.1.1.59884       ESTAB
```

Basic BGP Configuration

- BGP configuration components
- BGP configuration steps
- Verifying BGP sessions
- BGP tables and prefix advertisement
- How BGP receives, processes, and views routes

BGP Configuration Components

BGP router configuration requires the following components:

BGP session parameters - settings for establishing communication to the remote BGP neighbor including: ASN of the BGP peer, authentication, keepalive timers, and source and destination IP address settings for the session.

Address family initialization - initialized under the BGP router configuration mode. Network advertisement and summarization occur within the address family.

Activation of the address family on the BGP peer - must be activated for a BGP peer in order for BGP to initiate a session with that peer. The router's IP address is added to the neighbor table, and BGP attempts to establish a BGP session or accepts a BGP session initiated from the peer router.

BGP Configuration Steps

Step 1. Initialize the BGP process with the global command **router bgp** *as-number.*

Step 2. Statically configure the BGP router ID (RID) (optional). The dynamic RID allocation uses the highest IP of the loopback interfaces. If there is not a loopback interface, then the highest IP address of any active up interfaces becomes the RID. Statically configuring the BGP RID using the command **bgp router-id** *router-id* is a best practice. When the router ID changes, all BGP sessions reset and need to be reestablished.

Step 3. Configure the BGP neighbor's IP address and autonomous system number with the BGP router configuration command **neighbor** *ip-address* **remote-as** *as-number*

Step 4. Specify the source interface for the BGP session (Optional). It is important to understand the traffic flow of BGP packets between peers. The source IP address of the BGP packets still reflects the IP address of the outbound interface. When a BGP packet is received, the router correlates the source IP address of the packet to the IP address configured for that neighbor. If the BGP packet source does not match an entry in the neighbor table, the packet cannot be associated to a neighbor and is discarded. Specify the interface for the BGP session for a specific neighbor with the command **neighbor** *ip-address* **updatesource** *interface-id*

BGP Configuration Steps (Cont.)

Step 5. Enable BGP authentication (optional) using a Message Digest 5 (MD5) authentication hash. To enable BGP authentication, place the command **neighbor** *ip-address* **password** *password* under the neighbor session parameters.

Step 6. Modify the BGP timers (optional). BGP KEEPALIVE and UPDATE messages ensure that the BGP neighbor is established. The default hold timer requires that a packet be received every 3 minutes (180 seconds) to maintain the BGP session. By default, BGP sends a KEEPALIVE message to a BGP neighbor every 60 seconds. The BGP keepalive timer and hold timer can be set at the process level or per neighbor session. The BGP timers can be modified for a session with the command **neighbor** *ip-address* **timers** *keepalive holdtime [minimum-holdtime]*

Step 7. Initialize the address family with the BGP router configuration command **address-family** *afi safi*. Examples of AFIs are IPv4 and IPv6 and examples of SAFIs are unicast and multicast.

Step 8. Activate the address family for the BGP neighbor by using the BGP address family configuration command **neighbor** *ip-address* **activate**

IOS activates the *IPv4 address family* by default. The command **no bgp default ip4-unicast** disables the automatic activation.

Simple eBGP Topology

Example 11-2 demonstrates how to configure R1 and R2 using the IOS default and optional IPv4 AFI modifier CLI syntax.

- R1 is configured using the default IPv4 address family enabled.
- R2 disables IOS's default IPv4 address family and manually activates it for the specific neighbor 10.12.1.1.
- Authentication is enabled with the password of CISCOBGP.
- R1 sets the BGP hello timer to 10 seconds and the hold timer to 40 seconds.
- R2 sets the BGP hello timer to 15 seconds and the hold timer to 50 seconds.

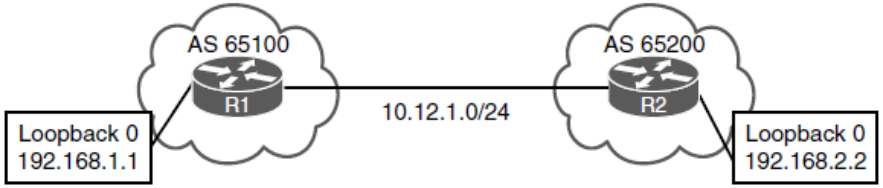

Figure 11-3 *Simple eBGP Topology*

Example 11-2 *BGP Configuration*

```
R1 (Default IPv4 Address-Family Enabled)
router bgp 65100
 neighbor 10.12.1.2 remote-as 65200
 neighbor 10.12.1.2 password CISCOBGP
 neighbor 10.12.1.2 timers 10 40
```

```
R2 (Default IPv4 Address-Family Disabled)
router bgp 65200
 no bgp default ipv4-unicast
 neighbor 10.12.1.1 remote-as 65100
 neighbor 10.12.1.2 password CISCOBGP
 neighbor 10.12.1.1 timers 15 50
 !
 address-family ipv4
  neighbor 10.12.1.1 activate
 exit-address-family
```

Verification of BGP Sessions

A BGP session is verified with the **show bgp** *afi safi* **summary** command.

Example 11-3 shows the output from the **show bgp ipv4 unicast summary** command. Notice that the BGP RID and table version are the first components shown. The Up/Down column indicates that the BGP session is up for over 5 minutes.

Earlier commands, such as **show ip bgp summary**, came out before MP-BGP and do not provide a structure for the current multiprotocol capabilities in BGP.

Example 11-3 *BGP IPv4 Session Summary Verification*

```
R1# show bgp ipv4 unicast summary
BGP router identifier 192.168.2.2, local AS number 65200
BGP table version is 1, main routing table version 1

Neighbor        V     AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down  State/PfxRcd
10.12.1.2       4  65200       8       9        1    0    0 00:05:23        0
```

Table 11-3 BGP Summary Fields

Field	Description
Neighbor	IP address of the BGP peer
V	BGP version used by the BGP peer
AS	Autonomous system number of the BGP peer
MsgRcvd	Count of messages received from the BGP peer
MsgSent	Count of messages sent to the BGP peer
TblVer	Last version of the BGP database sent to the peer
InQ	Number of messages queued to be processed from the peer
OutQ	Number of messages queued to be sent to the peer
Up/Down	Length of time the BGP session is established, or the current status if the session is not in an established state
State/PfxRcd	Current state of the BGP peer or the number of prefixes received from the peer

Verification of BGP Sessions (Cont.)

You can get BGP neighbor session state, timers, and other essential peering information by using the **show bgp** *afi safi* **neighbors** *ip-address* command, as shown in Example 11-4.

Notice that the BGP hold time has negotiated to 40 based on R1's session settings.

Example 11-4 *BGP IPv4 Neighbor Output*

```
R2# show bgp ipv4 unicast neighbors 10.12.1.1
! Output omitted for brevity

! The first section provides the neighbor's IP address, remote-as, indicates if
! the neighbor is 'internal' or 'external', the neighbor's BGP version, RID,
! session state, and timers.

BGP neighbor is 10.12.1.1, remote AS65100, external link
  BGP version 4, remote router ID 192.168.1.1
  BGP state = Established, up for 00:01:04
  Last read 00:00:10, last write 00:00:09, hold is 40, keepalive is 13 seconds
  Neighbor sessions:
    1 active, is not multisession capable (disabled)
! This second section indicates the capabilities of the BGP neighbor and
! address-families configured on the neighbor.
  Neighbor capabilities:
    Route refresh: advertised and received(new)
    Four-octets ASN Capability: advertised and received
    Address family IPv4 Unicast: advertised and received
    Enhanced Refresh Capability: advertised
    Multisession Capability:
    Stateful switchover support enabled: NO for session 1
  Message statistics:
    InQ depth is 0

! This section provides a list of the BGP packet types that have been received
! or sent to the neighbor router.
                         Sent       Rcvd
    Opens:                  1          1
    Notifications:          0          0
    Updates:                0          0
    Keepalives:             2          2
    Route Refresh:          0          0
    Total:                  4          3
  Default minimum time between advertisement runs is 0 seconds

! This section provides the BGP table version of the IPv4 Unicast address-
! family. The table version is not a 1-to-1 correlation with routes as multiple
! route change can occur during a revision change. Notice the Prefix Activity
! columns in this section.
```

Prefix Advertisement – BGP Tables

BGP uses three tables for maintaining the network paths and path attributes (PAs) for a prefix.

- **Adj-RIB-in -** Contains the Network Layer Reachability Information (NLRI) routes in original form (before inbound route policies are processed). The table is purged after all route policies are processed to save memory.
- **Loc-RIB -** Contains all the NLRI routes that originated locally or were received from other BGP peers. After NLRI routes pass the validity and next-hop reachability check, the BGP best-path algorithm selects the best NLRI for a specific prefix. The Loc-RIB table is the table used for presenting routes to the IP routing table.
- **Adj-RIB-out -** Contains the NLRI routes after outbound route policies have been processed.

You install network prefixes in the BGP Loc-RIB table with the **network** *network* **mask** *subnet-mask [route-map route-map-name]* command under the appropriate BGP address family configuration. The optional **route-map** provides a method to set specific BGP PAs when the prefix installs into the Loc-RIB table.

Prefix Advertisement

The BGP network statements identify a specific network prefix to be installed into the BGP table, known as the Loc-RIB table. The BGP process searches the global RIB for an exact network prefix match. The network prefix can be a connected network, a secondary connected network, or any route from a routing protocol.

After verifying that the network statement matches a prefix in the global RIB, the prefix is installed into the BGP Loc-RIB table. As the BGP prefix is installed into the Loc-RIB table, the following BGP PAs are set, depending on the RIB prefix type:

- **Connected network -** The next-hop BGP attribute is set to 0.0.0.0, the BGP origin attribute is set to i (for IGP), and the BGP weight is set to 32,768.
- **Static route or routing protocol -** The next-hop BGP attribute is set to the next-hop IP address in the RIB, the BGP origin attribute is set to i (for IGP), the BGP weight is set to 32,768, and the multi-exit discriminator (MED) is set to the IGP metric.

Prefix Advertisement (Cont.)

Figure 11-5, R1 has multiple routes learned from static routes, EIGRP, and OSPF. R3's loopback was learned through EIGRP, R4's loopback is reached using a static route, and R5's loopback is learned from OSPF.

All the routes in R1's routing table can be advertised into BGP, regardless of the source routing protocol.

Example 11-5 demonstrates the configuration for advertising the loopback interface of R1, R3, and R4 using network statements on R1. R5's loopback interface is injected into BGP through redistribution of OSPF into BGP.

R2's configuration resides under **address-family ipv4 unicast** because the default IPv4 unicast address family has been disabled.

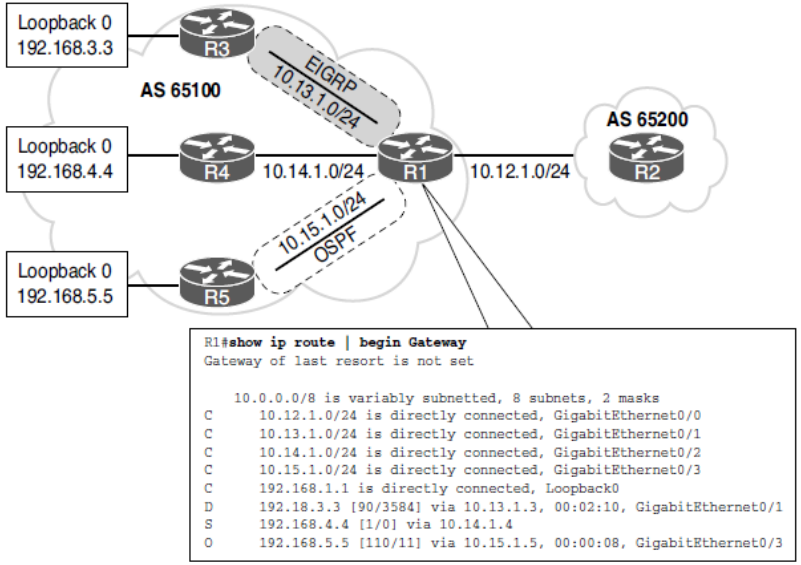

Figure 11-5 *Multiple BGP Route Sources*

Example 11-5 *Configuration for Advertising Non-Connected Routes*

```
R1
router bgp 65100
 network 10.12.1.10 mask 255.255.255.0
 network 192.168.1.1 mask 255.255.255.255
 network 192.168.3.3 mask 255.255.255.255
 network 192.168.4.4 mask 255.255.255.255
 redistribute ospf 1
```

```
R2
router bgp 65200
 address-family ipv4 unicast
  network 10.12.1.0 mask 255.255.255.0
  network 192.168.2.2 mask 255.255.255.255
```

Prefix Advertisement (Cont.)

Not every prefix in the Loc-RIB table is advertised or installed into the global RIB. When a route is received from a peer, BGP performs the following route processing steps, shown in Figure 11-6:

Step 1. Store the route in the Adj-RIB-In table in the original state and apply the inbound route policy, based on the neighbor from which it was received.

Step 2. Update the Loc-RIB table with the latest entry. The Adj-RIB-In table is cleared to save memory.

Step 3. Pass a validity check to verify that the route is valid and that the next-hop address is resolvable in the global RIB. If the route fails, the route remains in the Loc-RIB table but is not processed further.

Step 4. Identify the BGP best path and pass only the best path and its path attributes to step 5.

Step 5. Install the best-path route into the global RIB, process the outbound route policy, store the non-discarded routes in the Adj-RIB-Out table, and advertise to BGP peers.

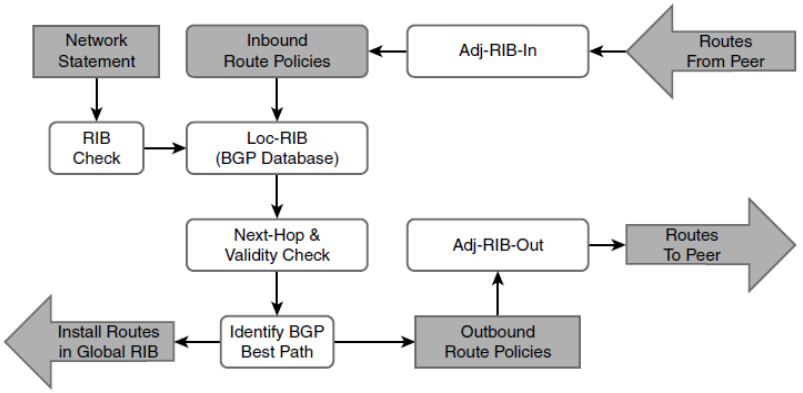

Figure 11-6 *BGP Database Processing*

Prefix Advertisement – BGP Table

The **show bgp** *afi safi* command displays the contents of the BGP database (Loc-RIB table). Every entry in the BGP Loc-RIB table contains at least one path but could contain multiple paths for the same network prefix. Example 11-6 displays the BGP table on R1, which contains locally generated routes and routes from R2.

On R1, the next hop matches the next hop learned from the RIB, AS_Path is blank, and the origin code is IGP (for routes learned from the network statement) or incomplete (redistributed). The metric is carried over from R3's and R5's IGP routing protocols and is indicated as MED (Metric).

R2 learns the routes strictly from eBGP and sees only MED (Metric) and the origin codes (i, ?).

Example 11-6 *BGP Table of Routes from Multiple Sources*

```
R1# show bgp ipv4 unicast
BGP table version is 9, local router ID is 192.168.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 *>  10.12.1.0/24     0.0.0.0                  0         32768 i
 *                    10.12.1.2                0             0 65200 i
 *>  10.15.1.0/24     0.0.0.0                  0         32768 ?
 *>  192.168.1.1/32   0.0.0.0                  0         32768 i
 *>  192.168.2.2/32   10.12.1.2                0             0 65200 i
 ! The following route comes from EIGRP and uses a network statement
 *>  192.168.3.3/32   10.13.1.3             3584         32768 i
! The following route comes from a static route and uses a network statement
 *>  192.168.4.4/32   10.14.1.4               0         32768 i
! The following route was redistributed from OSPF
 *>  192.168.5.5/32   10.15.1.5              11         32768 ?
```

```
R2# show bgp ipv4 unicast | begin Network
     Network          Next Hop            Metric LocPrf Weight Path
 *   10.12.1.0/24     10.12.1.1                0             0 65100 i
 *>                   0.0.0.0                  0         32768 i
 *>  10.15.1.0/24     10.12.1.1                0             0 65100 ?
 *>  192.168.1.1/32   10.12.1.1                0             0 65100 i
 *>  192.168.2.2/32   0.0.0.0                  0         32768 i
 *>  192.168.3.3/32   10.12.1.1             3584             0 65100 i
 *>  192.168.4.4/32   10.12.1.1                0             0 65100 i
 *>  192.168.5.5/32   10.12.1.1               11             0 65100 ?
```

Prefix Advertisement – BGP Table Fields

Table 11-4 explains the fields of output when displaying the BGP table.

Example 11-6 *BGP Table of Routes from Multiple Sources*

```
R1# show bgp ipv4 unicast
BGP table version is 9, local router ID is 192.168.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 *>  10.12.1.0/24     0.0.0.0                  0         32768 i
 *                    10.12.1.2                0             0 65200 i
 *>  10.15.1.0/24     0.0.0.0                  0         32768 ?
 *>  192.168.1.1/32   0.0.0.0                  0         32768 i
 *>  192.168.2.2/32   10.12.1.2                0             0 65200 i
 ! The following route comes from EIGRP and uses a network statement
 *>  192.168.3.3/32   10.13.1.3             3584         32768 i
! The following route comes from a static route and uses a network statement
 *>  192.168.4.4/32   10.14.1.4               0         32768 i
! The following route was redistributed from OSPF
 *>  192.168.5.5/32   10.15.1.5              11         32768 ?
```

Table 11-4 BGP Table Fields

Field	Description
Network	A list of the network prefixes installed in BGP. If multiple NLRI routes exist for the same prefix, only the first prefix is identified, and others leave a blank space. Valid NLRI routes are indicated by the *. The NLRI selected as the best path is indicated by an angle bracket (>).
Next Hop	A well-known mandatory BGP path attribute that defines the IP address for the next hop for that specific NLRI.
Metric	Multiple-exit discriminator (MED), an optional non-transitive BGP path attribute used in the BGP best-path algorithm for that specific NLRI.
LocPrf	Local preference, a well-known discretionary BGP path attribute used in the BGP best-path algorithm for that specific NLRI.
Weight	A locally significant Cisco-defined attribute used in the BGP best-path algorithm for that specific NLRI.
Path and Origin	AS_Path, a well-known mandatory BGP path attribute used for loop prevention and in the BGP best-path algorithm for that specific NLRI. Origin, a well-known mandatory BGP path attribute used in the BGP best-path algorithm. The value *i* represents an IGP, *e* is for EGP, and *?* is for a route that was redistributed into BGP.

Prefix Advertisement – BGP Prefix Attributes

The **show bgp** *afi safi network* command displays all the paths for a specific route and the BGP path attributes for that route. Example 11-7 displays the number of paths and the best path for the 10.12.1.0/24 network.

Table 11-5 explains the output provided in Example 11-7 and the correlation of each part of the output to BGP attributes. Some of the BGP path attributes may change, depending on the BGP features used.

Example 11-7 *Viewing Explicit BGP Routes and Path Attributes*

```
R1# show bgp ipv4 unicast 10.12.1.0
BGP routing table entry for 10.12.1.0/24, version 2
Paths: (2 available, best #2, table default)
  Advertised to update-groups:
     2
  Refresh Epoch 1
  65200
    10.12.1.2 from 10.12.1.2 (192.168.2.2)
      Origin IGP, metric 0, localpref 100, valid, external
      rx pathid: 0, tx pathid: 0
  Refresh Epoch 1
  Local
    0.0.0.0 from 0.0.0.0 (192.168.1.1)
      Origin IGP, metric 0, localpref 100, weight 32768, valid, sourced, local, best
      rx pathid: 0, tx pathid: 0x0
```

Table 11-5 BGP Prefix Attributes

Output	Description
Paths: (2 available, best #2)	Provides a count of BGP paths in the BGP Loc-RIB table and identifies the path selected as the BGP best path. All the paths and BGP attributes are listed after this.
Advertised to update-groups	Identifies whether the prefix was advertised to a BGP peer. BGP neighbors are consolidated into BGP update groups. If a route is not advertised then *Not advertised to any peer* is displayed.
65200 (1st path) Local (2nd path)	This is the AS_Path for the NLRI as it was received or whether the prefix was locally advertised.
10.1.12.2 from 10.1.12.2 (192.168.2.2)	The first entry lists the IP address of the eBGP edge peer. The 'from' field lists the IP address of the iBGP router that received this route from the eBGP edge peer. (In this case, the route was learned from an eBGP edge peer, so the address is the eBGP edge peer.) Expect this field to change when an external route is learned from an iBGP peer. The number in parentheses is the BGP identifier (RID) for that node.
Origin IGP	Origin is the BGP well-known mandatory attribute that states the mechanism for advertising this route. In this instance, it is an internal route.
metric 0	Displays the optional non-transitive BGP attribute MED, also known as the BGP metric.
localpref 100	Displays the well-known discretionary BGP attribute Local Preference.
valid	Displays the validity of this path.
External (1st path) Local (2nd path)	Displays how the route was learned: internal, external, or local.

Prefix Advertisement – Adj-RIB-OUT Table

The Adj-RIB-Out table is a unique table maintained for each BGP peer. It enables a network engineer to view routes advertised to a specific router. The **show bgp** *afi safi* **neighbors** *ip-address* **advertised-routes** command displays the contents of the Adj-RIB-Out table for a neighbor.

Example 11-8 shows the Adj-RIB-Out entries specific to each neighbor. The next-hop address reflects the local router's BGP table and is changed as the route advertises to the peer.

Example 11-8 *Neighbor-Specific View of the Adj-RIB-OUT Table*

```
R1# show bgp ipv4 unicast neighbors 10.12.1.2 advertised-routes
! Output omitted for brevity
     Network          Next Hop            Metric LocPrf Weight Path
 *> 10.12.1.0/24     0.0.0.0                  0         32768 i
 *>  10.15.1.0/24    0.0.0.0                  0         32768 ?
 *>  192.168.1.1/32  0.0.0.0                  0         32768 i
 *>  192.168.3.3/32  10.13.1.3              3584         32768 i
 *>  192.168.4.4/32  10.14.1.4               0         32768 i
 *>  192.168.5.5/32  10.15.1.5              11         32768 ?

Total number of prefixes 6
```

```
R2# show bgp ipv4 unicast neighbors 10.12.1.1 advertised-routes
! Output omitted for brevity
     Network          Next Hop            Metric LocPrf Weight Path
*> 10.12.1.0/24     0.0.0.0                  0         32768 i
*> 192.168.2.2/32   0.0.0.0                  0         32768 i

Total number of prefixes 2
```

Prefix Advertisement – Verifying Routes

The **show bgp ipv4 unicast summary** command is also used to verify the exchange of NLRI routes between nodes, as shown in Example 11-9.

Example 11-9 *BGP Summary with Prefixes*

```
R1# show bgp ipv4 unicast summary
! Output omitted for brevity
Neighbor        V           AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down  State/PfxRcd
10.12.1.2       4        65200      11      10        9    0    0 00:04:56            2
```

You display the BGP routes in the global IP routing table (RIB) by using the **show ip route bgp** command. In Example 11-10 the prefixes are from an eBGP session and have an AD of 20, and no metric is present.

Example 11-10 *Displaying BGP Routes in an IP Routing Table*

```
R1# show ip route bgp | begin Gateway
Gateway of last resort is not set

      192.168.2.0/32 is subnetted, 1 subnets
B        192.168.2.2 [20/0] via 10.12.1.2, 00:06:12
```

Understanding BGP Session Types and Behavior

- BGP sessions are always point-to-point between two routers and are categorized into two types:
  - Internal BGP (iBGP) - Sessions established with an iBGP routers that are in the same AS or that participate in the same confederation
  - External BGP (eBGP) - Sessions established with a BGP router in a different AS

iBGP

BGP sessions are always point-to-point between two routers and are categorized into two types:

Internal BGP (iBGP) - Sessions established with iBGP routers that are in the same AS or that participate in the same confederation. The need for BGP within an AS typically occurs when multiple routing policies are needed or when transit connectivity is provided between autonomous systems. Some of BGPs security measures are lowered in iBGP as compared to eBGP. iBGP prefixes are assigned an administrative distance of 200.

External BGP (eBGP) - Sessions established with a BGP router that are in a different AS. eBGP prefixes are assigned an AD of 20.

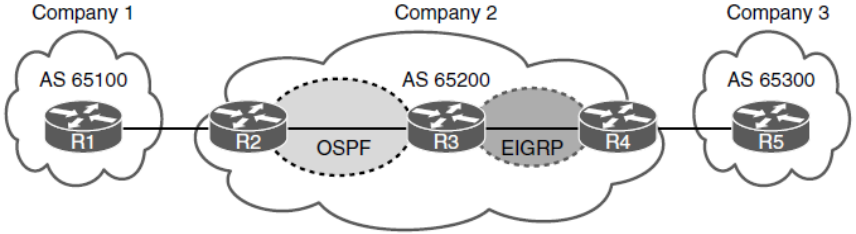

Figure 11-7 *AS 65200 Provides Transit Connectivity*

In Figure 11-7, AS 65200 provides transit connectivity to AS 65100 and AS 65300. AS 65100 connects at R2, and AS 65300 connects at R4.

iBGP (Cont.)

In Figure 11-8, if R2 forms an iBGP session with R4, R3 would not know where to route traffic from AS 65100 or AS 65300, because R3 does not have the route forwarding information for networks in AS 65100 or AS 65300. Redistributing the BGP table into an IGP is not a viable solution for these reasons:

Scalability - The internet has 800,000+ IPv4 network prefixes. IGPs cannot scale to that level of routes.

Custom routing: IGP link-state and distance vector routing protocols use metric as the primary method for route selection. BGP uses BGP path attributes to manipulate the path for a specific prefix (NLRI). A BGP path might not be optimal from an IGP's perspective.

Path attributes: All the BGP path attributes cannot be maintained within IGP protocols. Only BGP is capable of maintaining the path attribute as the prefix is advertised from one edge of the AS to the other edge.

Establishing iBGP sessions between all the routers in a full mesh (R2, R3, and R4 in this case) allows for proper forwarding between autonomous systems.

Note: Service providers provide transit connectivity. Enterprise organizations are consumers and should not provide transit connectivity between autonomous systems across the internet.

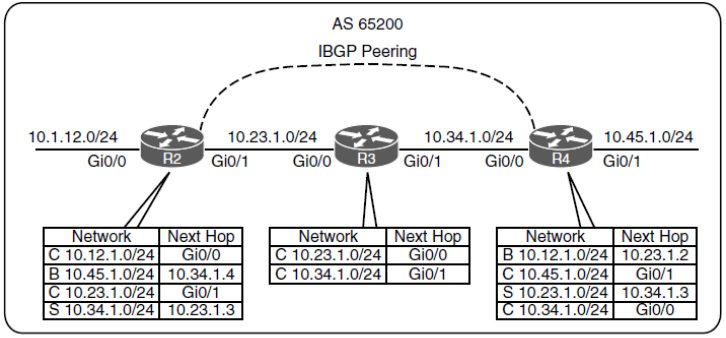

Figure 11-8 *iBGP Prefix Advertisement Behavior*

iBGP Full Mesh Requirement

- iBGP peers do not prepend their ASN to AS_Path because the NLRI would fail the validity check and would not install the prefix into the IP routing table.
- No other method exists for detecting loops with iBGP sessions, and RFC 4271 prohibits the advertisement of an NLRI received from an iBGP peer to another iBGP peer.
- All BGP routers in a single AS must be fully meshed to provide a complete loop free routing table and prevent traffic black-holing.

Figure 11-9, R1 has an iBGP session with R2, and R2 has an iBGP session with R3. R1 advertises the 10.1.1.0/24 prefix to R2, which is processed and inserted into R2's BGP table. R2 does not advertise the 10.1.1.0/24 NLRI to R3 because it received the prefix from an iBGP peer.

To resolve this issue, R1 must form a multi-hop iBGP session so that R3 can receive the 10.1.1.0/24 prefix directly from R1. R1 connects to R3's 10.23.1.3 IP address, and R3 connects to R1's 10.12.1.1 IP address. R1 and R3 need a static route to the remote peering link, or R2 needs to advertise the 10.12.1.0/24 and 10.23.1.0/24 networks into BGP.

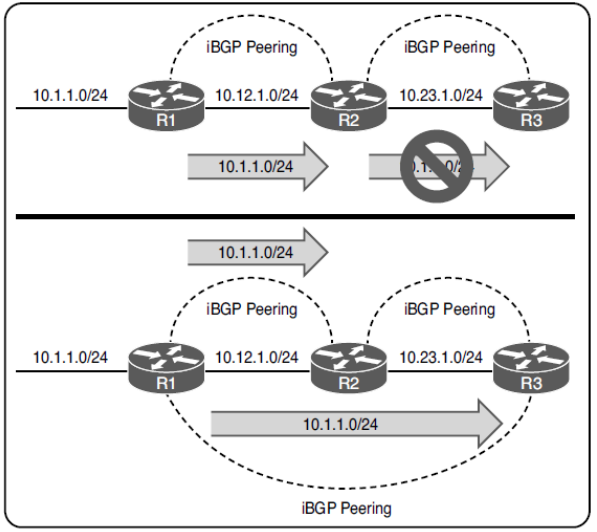

Figure 11-9 *iBGP Prefix Advertisement Behavior*

Peering Using Loopback Addresses

BGP sessions are sourced by the IP address of the outbound interface toward the BGP peer by default. In Figure 11-10, R1, R2, and R3 are a full mesh of iBGP sessions peered by transit links.

If there is a failure of the 10.13.1.0/24 network, R3's BGP session with R1 times out and terminates. R3 loses connectivity to the 10.1.1.0/24 network, even though R1 and R3 could communicate through R2 (through a multi-hop path). iBGP does not advertise routes learned from another iBGP peer. You can use either of two solutions to overcome the link failure:

- Add a second link between each pair of routers (so that three links become six links) and establish two BGP sessions between each pair of routers.
- Configure an IGP on the router's transit links, advertise loopback interfaces into the IGP, and then configure the BGP neighbors to establish a session to the remote router's loopback address. This method is preferable of the two methods because the loopback interface is virtual and always stays up.

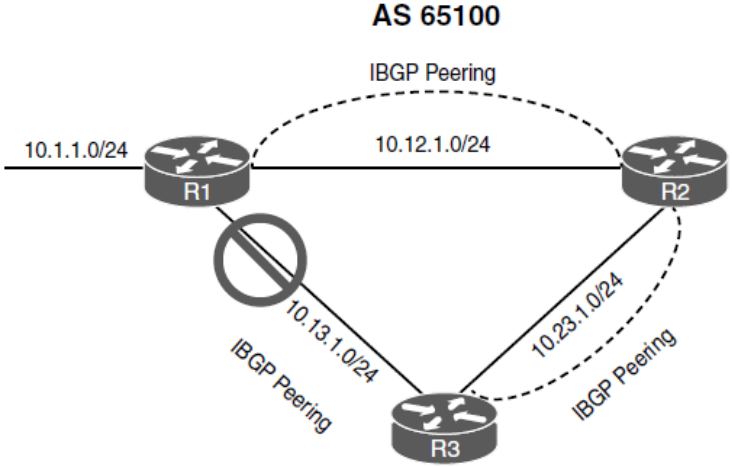

Figure 11-10 *Link Failure in a Full-Mesh iBGP Topology*

Peering Using Loopback Addresses (Cont.)

It is not enough to update the BGP configuration to connect the session to the remote router's loopback IP address. The source IP address of the BGP packets still reflects the IP address of the outbound interface, not the loopback address. When the BGP packet source does not match an entry in the neighbor table, the packet cannot be associated to a neighbor and is discarded.

You can statically set the source address to an interface's primary IP address using the **neighbor** *ip-address* **update-source** *interface-id* command. Example 11-11 shows the configuration for R1 and R2 from the Figure 11-11 to peer using loopback interfaces. R1 has the default IPv4 address family enabled, and R2 does not.

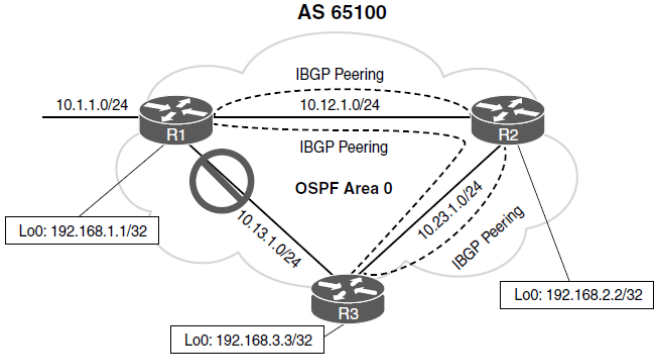

Figure 11-11 *Link Failure with iBGP Sessions on Loopback Interfaces*

Example 11-11 *BGP Configuration Source from Loopback Interfaces*

```
R1 (Default IPv4 Address-Family Enabled)
router ospf 1
 network 10.12.0.0 0.0.255.255 area 0
 network 10.13.0.0 0.0.255.255 area 0
 network 192.168.1.1 0.0.0.0 area 0
!
router bgp 65100
 network 10.1.1.0 mask 255.255.255.0
 neighbor 192.168.2.2 remote-as 100
 neighbor 192.168.2.2 update-source Loopback0
 neighbor 192.168.3.3 remote-as 100
 neighbor 192.168.3.3 update-source Loopback0
 !
 address-family ipv4
  neighbor 192.168.2.2 activate
  neighbor 192.168.3.3 activate
```

```
R2 (Default IPv4 Address-Family Disabled)
router ospf 1
 network 10.0.0.0 0.255.255.255 area 0
 network 192.168.2.2 0.0.0.0 area 0
!
router bgp 65100
 no bgp default ipv4-unicast
 neighbor 192.168.1.1 remote-as 100
 neighbor 192.168.1.1 update-source Loopback0
 neighbor 192.168.3.3 remote-as 100
 neighbor 192.168.3.3 update-source Loopback0
 !
 address-family ipv4
  neighbor 192.168.1.1 activate
  neighbor 192.168.3.3 activate
```

eBGP

eBGP peerings are the core component of BGP on the internet. eBGP involves the exchange of network prefixes between autonomous systems. The following behaviors are different on eBGP sessions than on iBGP sessions:

- The time-to-live (TTL) on eBGP packets is set to 1. BGP packets drop in transit if a multi-hop BGP session is attempted. The TTL on iBGP packets is set to 255, which allows for multi-hop sessions.
- The advertising router modifies the BGP next hop to the IP address sourcing the BGP connection.
- The advertising router prepends its ASN to the existing AS_Path.
- The receiving router verifies that the AS_Path does not contain an ASN that matches the local routers. BGP discards the NLRI if it fails the AS_Path loop-prevention check.

The configurations for eBGP and iBGP sessions are fundamentally the same except that the ASN in the remote-as statement is different from the ASN defined in the BGP process.

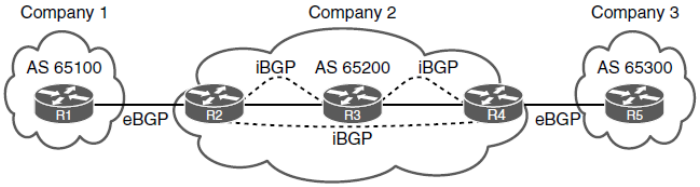

Figure 11-12 *eBGP and iBGP Sessions*

Figure 11-12 demonstrates the eBGP and iBGP sessions that would be needed between the routers to allow connectivity between AS 65100 and AS 65300. Notice that AS 65200 R2 establishes an iBGP session with R4 to overcome the loop-prevention behavior of iBGP learned routes, as explained earlier.

eBGP and iBGP Topologies

Combining eBGP sessions with iBGP sessions can cause confusion in terminology and concepts. Figure 11-13 provides a reference topology for clarification of eBGP and iBGP concepts.

R1 and R2 form an eBGP session, R3 and R4 form an eBGP session as well, and R2 and R3 form an iBGP session. R2 and R3 are iBGP peers and follow the rules of iBGP advertisement, even if the routes are learned from an eBGP peer.

As an eBGP prefix is advertised to an iBGP neighbor, issues may arise with the NLRI passing the validity check and next-hop reachability check, preventing advertisements to other BGP peers. The most common issue involves the failure of the next-hop accessibility check. iBGP peers do not modify the next-hop address if the NLRI has a next-hop address other than 0.0.0.0. The next-hop address must be resolvable in the global RIB in order for it to be valid and advertised to other BGP peers.

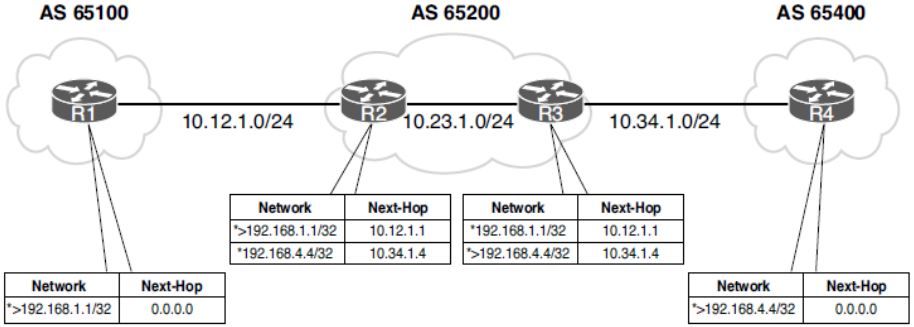

Figure 11-13 *eBGP and iBGP Topology*

eBGP and iBGP Topologies (Cont.)

In this scenario, R1 and R4 have advertised their loopback interfaces into BGP: 192.168.1.1/32 and 192.168.4.4/32, respectively. Figure 11-13 displays the BGP table for all four routers. Notice that the BGP best-path symbol (>) is missing for the 192.168.4.4/32 prefix on R2 and for the 192.168.1.1/32 prefix on R3.

R1's BGP table is missing the 192.168.4.4/32 prefix because the prefix did not pass R2's next hop accessibility check. Example 11-13 shows the BGP attributes on R3 for the 192.168.1.1/32 prefix. Notice that the prefix is not advertised to any peer because the next-hop is inaccessible.

Example 11-13 *BGP Path Attributes for 192.168.1.1/32*

```
R3# show bgp ipv4 unicast 192.168.1.1
BGP routing table entry for 192.1681.1/32, version 2
Paths: (1 available, no best path)
  Not advertised to any peer
  Refresh Epoch 1
  65100
    10.12.1.1 (inaccessible) from 10.23.1.2 (192.168.2.2)
      Origin IGP, metric 0, localpref 100, valid, internal
```

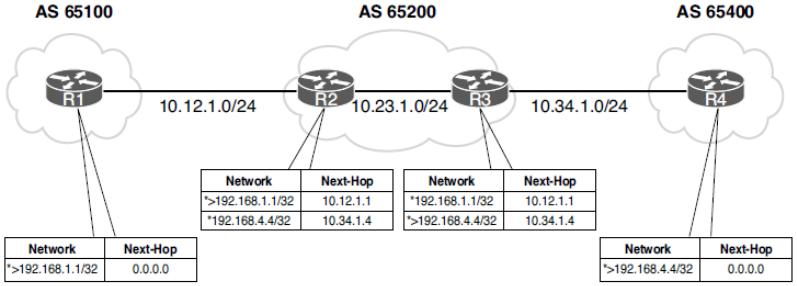

Figure 11-13 *eBGP and iBGP Topology*

To correct the issue, the peering links 10.12.1.0/24 and 10.34.1.0/24 need to be in both R2's and R3's routing tables, using either of these techniques:

- IGP advertisement (Remember to use the passive interface to prevent an accidental adjacency forming.)
- Advertise the networks into BGP

eBGP and iBGP Topologies (Cont.)

Figure 11-14 displays the topology with both transit links advertised into BGP. Notice that this time, all four prefixes are valid, with a BGP best path selected.

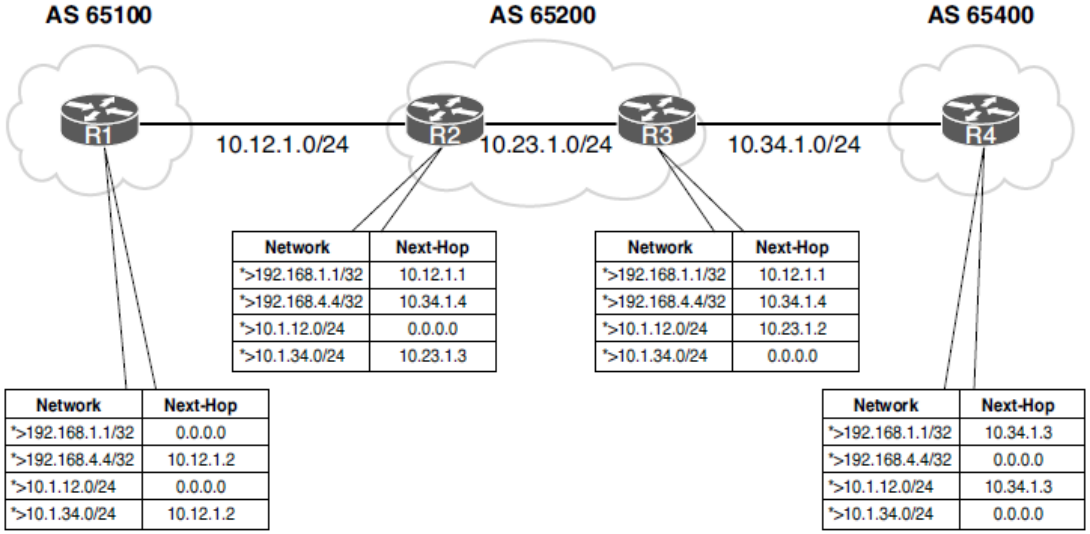

Figure 11-14 *eBGP and iBGP Topology After Advertising Peering Links*

eBGP and iBGP Topologies (Cont.)

Another technique to ensure that the next-hop address check passes without advertising peering networks into a routing protocol involves the modification of the next-hop address. The next-hop IP address can be modified on inbound or outbound neighbor routing policies.

The next-hop-self feature modifies the next-hop address in the NLRI for external BGP prefixes in the IP address sourcing the BGP session.

The **neighbor** *ip-address* **next-hop-self [all]** command is used for each neighbor under the address family configuration. The next-hop-self feature does not modify the next-hop address for iBGP prefixes by default. IOS nodes can append the optional **all** keyword, which modifies the next-hop address on iBGP prefixes, too.

Example 11-14 shows the configuration for R2 and R3 so that the eBGP peer links do not need to be advertised into BGP.

Example 11-14 *BGP Configuration Source for Next-Hop-Self*

```
R2 (Default IPv4 Address-Family Enabled)
router bgp 65200
 neighbor 10.12.1.1 remote-as 65100
 neighbor 10.23.1.3 remote-as 65200
 neighbor 10.23.1.3 next-hop-self
```

```
R3 (Default IPv4 Address-Family Disabled)
router bgp 65200
 no bgp default ipv4-unicast
 neighbor 10.23.1.2 remote-as 65200
 neighbor 10.34.1.4 remote-as 65400
 !
 address-family ipv4
  neighbor 10.23.1.2 activate
  neighbor 10.23.1.2 next-hop-self
  neighbor 10.34.1.4 activate
```

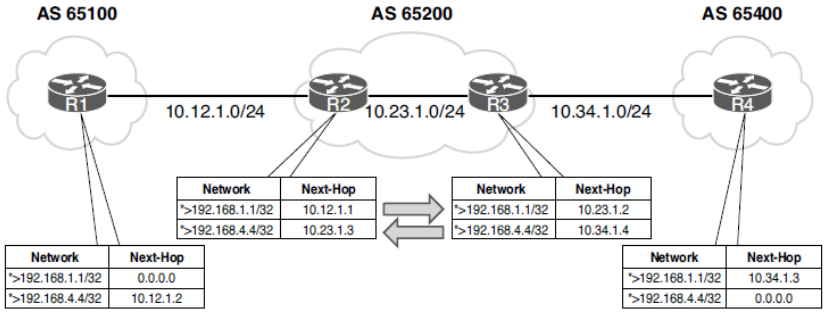

Figure 11-15 *eBGP and iBGP Topology with next-hop-self*

iBGP Scalability Enhancements

The inability of BGP to advertise a prefix learned from one iBGP peer to another iBGP peer leads to scalability issues in an AS. The formula n(n − 1)/2 provides the number of sessions required, where n represents the number of routers. A full mesh topology of 10 routers requires 45 sessions.

Route Reflectors - RFC 1966 introduces the idea that an iBGP peering can be configured so that it reflects routes to another iBGP peer. The router that is reflecting routes is known as a route reflector (RR), and the router that is receiving reflected routes is a route reflector client. Route reflectors and route reflection involve three basic rules:

- **Rule 1 -** If an RR receives an NLRI from a non-RR client, the RR advertises the NLRI to an RR client. It does not advertise the NLRI to a non-RR client.
- **Rule 2 -** If an RR receives an NLRI from an RR client, it advertises the NLRI to RR clients and non-RR clients. Even the RR client that sent the advertisement receives a copy of the route, but it discards the NLRI because it sees itself as the route originator.
- **Rule 3 -** If an RR receives a route from an eBGP peer, it advertises the route to RR clients and non-RR clients.

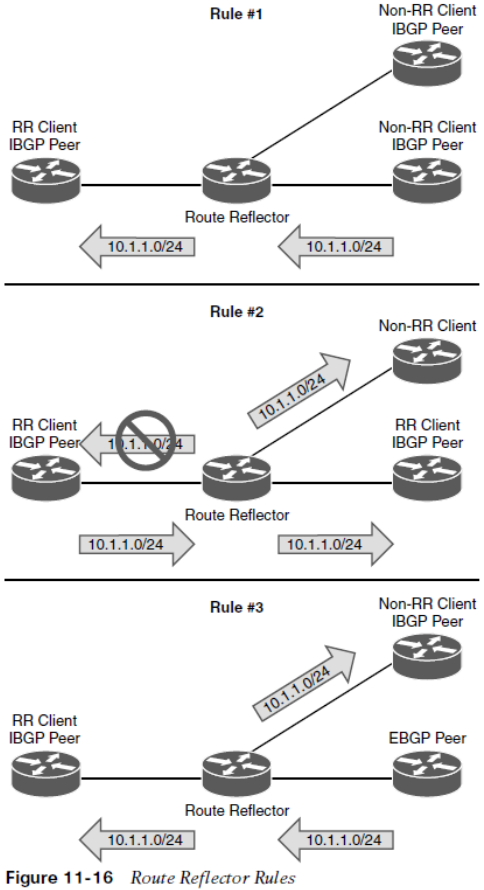

Figure 11-16 *Route Reflector Rules*

Route Reflector Configuration

Only route reflectors are aware of the change in behavior because no additional BGP configuration is performed on route reflector clients. BGP route reflection is specific to each address family. The **neighbor** *ip-address* **route-reflector-client** command is used under the neighbor address family configuration.

Figure 11-17 shows a simple iBGP topology. R1 is a route reflector client to R2, and R4 is a route reflector client to R3. R2 and R3 have a normal iBGP peering.

Example 11-15 shows the BGP configuration for R1, R2, R3, and R4. R1 and R2 are configured with the default IPv4 address family enabled, and R3 and R4 have the default IPv4 address family disabled. The **neighbor** *ip-address* **route-reflector-client** is configured only on R2 and R3. R1 explicitly advertises the 10.1.1.10/24 network with a network statement.

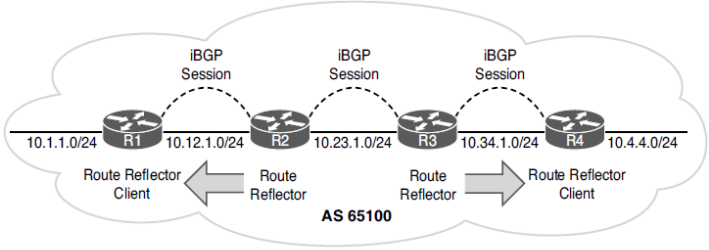

Figure 11-17 *Route Reflector Topology*

Example 11-15 *BGP Configuration Source from Loopback Interfaces*

```
R1 (Default IPv4 Address-Family Enabled)
router bgp 65100
 network 10.1.1.0 mask 255.255.255.0
 redistribute connected
 neighbor 10.12.1.2 remote-as 65100
```

```
R2 (Default IPv4 Address-Family Enabled)
router bgp 65100
 redistribute connected
 neighbor 10.12.1.1 remote-as 65100
 neighbor 10.12.1.1 route-reflector-client
 neighbor 10.23.1.3 remote-as 65100
```

```
R3 (Default IPv4 Address-Family Disabled)
router bgp 65100
 no bgp default ipv4-unicast
 neighbor 10.23.1.2 remote-as 65100
 neighbor 10.34.1.4 remote-as 65100
 !
address-family ipv4
  redistribute connected
  neighbor 10.23.1.2 activate
  neighbor 10.34.1.4 activate
  neighbor 10.34.1.4 route-reflector-client
```

```
R4 (Default IPv4 Address-Family Disabled)
router bgp 65100
 no bgp default ipv4-unicast
 neighbor 10.34.1.3 remote-as 65100
 !
 address-family ipv4
  neighbor 10.34.1.3 activate
 exit-address-family
```

Route Reflector Configuration (Cont.)

Figure 11-18 shows the topology with the route reflector and route reflector client roles to demonstrate the rules of a route reflector in action.

R1 advertises the 10.1.1.0/24 prefix to R2 as a normal iBGP advertisement. R2 receives and advertises the 10.1.1.0/24 prefix using the route reflector rule 2 as just explained to R3 (a non-route reflector client). R3 receives and advertises the 10.1.1.0/24 using the route reflector rule 1 as explained to R4 (a route reflector client).

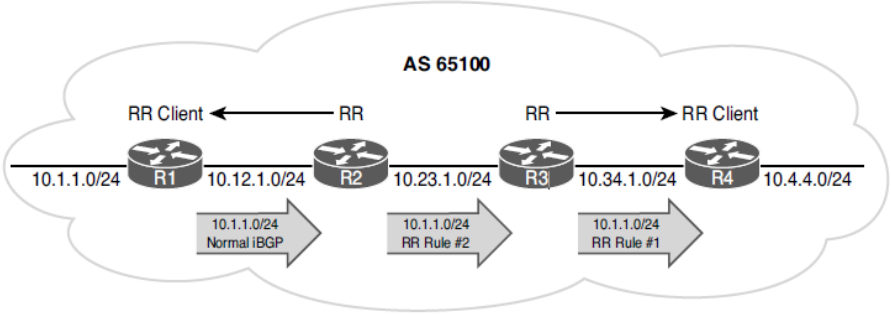

Figure 11-18 *Route Reflector Rules in a Topology*

Example 11-16 *BGP Configuration Source from Loopback Interfaces*

```
R1# show bgp ipv4 unicast | i Network|10.1.1
     Network          Next Hop            Metric LocPrf Weight Path
 *>  10.1.1.0/24      0.0.0.0                  0         32768 i

R2# show bgp ipv4 unicast | i Network|10.1.1
     Network          Next Hop            Metric LocPrf Weight Path
 *>i 10.1.1.0/24      10.12.1.1                0    100      0 i

R3# show bgp ipv4 unicast | i Network|10.1.1
     Network          Next Hop            Metric LocPrf Weight Path
 *>i 10.1.1.0/24      10.12.1.1                0    100      0 i

R4# show bgp ipv4 unicast | i Network|10.1.1
     Network          Next Hop            Metric LocPrf Weight Path
 *>i 10.1.1.0/24      10.12.1.1                0    100      0 i
```

Example 11-16 shows the BGP table for the 10.1.1.0/24 prefix. Notice that the next-hop IP address changes upon the route's installation into R2's BGP table, but it remains the same on R2, R3, and R4.

Notice the i immediately after the best path indicator (>) on R2, R3, and R4. This indicates that the prefix is learned through iBGP.

Loop Prevention in Route Reflectors

When RFC 1966 was drafted, two other BGP route reflector–specific attributes were added to prevent loops.

ORIGINATOR_ID: This optional non-transitive BGP attribute is created by the first route reflector and sets the value to the RID of the router that injected/advertised the route into the AS. If the ORIGINATOR-ID is already populated on an NLRI, it should not be overwritten. If a router receives an NLRI with its RID in the Originator attribute, the NLRI is discarded.

CLUSTER_LIST: This non-transitive BGP attribute is updated by the route reflector. This attribute is appended (not overwritten) by the route reflector with its cluster ID. By default, this is the BGP identifier. If a route reflector receives an NLRI with its cluster ID in the Cluster List attribute, the NLRI is discarded.

Example 11-17 shows all the BGP path attributes for the prefix 10.1.1.0/24 on R4. Notice that the originator and cluster fields are populated appropriately for the prefix.

Example 11-17 *Route Reflector Originator ID and Cluster List Attributes*

```
R4# show bgp ipv4 unicast 10.1.1.0/24
! Output omitted for brevity
Paths: (1 available, best #1, table default)
  Refresh Epoch 1
  Local
    10.12.1.1 from 10.34.1.3 (192.168.3.3)
      Origin IGP, metric 0, localpref 100, valid, internal, best
      Originator: 192.168.1.1, Cluster list: 192.168.3.3, 192.168.2.2
```

Confederations

RFC 3065 introduced the concept of BGP confederations as an alternative solution to the iBGP full mesh scalability issues shown earlier. A confederation consists of member ASs that combine into a larger AS known as an AS confederation.

Figure 11-19 demonstrates a BGP confederation with the confederation identifier AS200. R3 provides route reflection in member AS 65100.

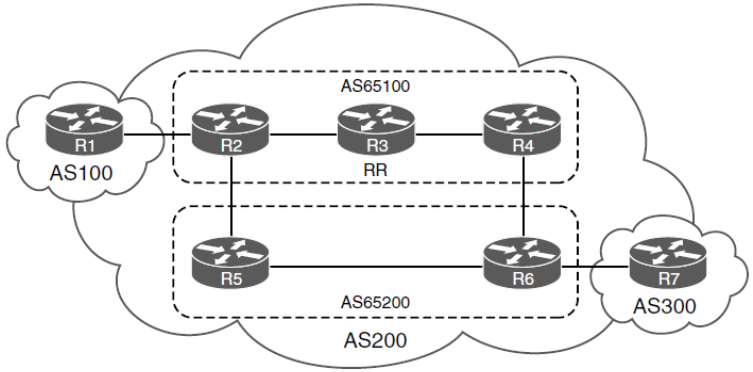

Figure 11-19 *Sample BGP Confederations Topology*

Follow these steps to configure a BGP confederation:

Step 1. Initialize the BGP process with the global command **router bgp** *member-asn*.

Step 2. Identify the BGP confederations with the command **bgp confederation identifier** *as-number*.

Step 3. On routers that directly peer with another member AS, identify the peering member AS with the command **bgp confederation peers** *member-asn*.

Step 4. Configure BGP confederation members as normal and then following the normal BGP configuration guidelines for the remaining configuration.

Understanding BGP Session Types and Behavior
Confederations (Cont.)

Example 11-18 Shows BGP session configuration for a confederations. R1 and R7 are not aware of the confederation and peer with R2 and R6 as though they were members of AS 200. Notice that R3 does not need the command bgp confederation peers because it is not peering with another member AS.

Confederations share behaviors from both iBGP sessions and eBGP sessions but have the following differences:

The AS_Path attribute contains a subfield called AS_CONFED_SEQUENCE. AS_CONFED_SEQUENCE is displayed in parentheses before any external ASNs in AS_Path. As the route passes from member AS to member AS, AS_CONFED_SEQUENCE is appended to contain the member AS ASNs. The AS_CONFED_SEQUENCE attribute is used to prevent loops but is not used (counted) when choosing the shortest AS_Path.

Example 11-18 *BGP Confederation Configuration*

R1
```
router bgp 100
 neighbor 10.12.1.2 remote-as 200
```

R2
```
router bgp 65100
 bgp confederation identifier 200
 bgp confederation peers 65200
neighbor 10.12.1.1 remote-as 100
 neighbor 10.23.1.3 remote-as 65100
 neighbor 10.25.1.5 remote-as 65200
```

R3
```
router bgp 65100
 bgp confederation identifier 200
 neighbor 10.23.1.2 remote-as 65100
 neighbor 10.23.1.2 route-reflector-client
 neighbor 10.34.1.4 remote-as 65100
 neighbor 10.34.1.4 route-reflector-client
```

R4
```
router bgp 65100
 bgp confederation identifier 200
 bgp confederation peers 65200
 neighbor 10.34.1.3 remote-as 65100
 neighbor 10.46.1.6 remote-as 65200
```

R5
```
router bgp 65200
 bgp confederation identifier 200
 bgp confederation peers 65100
 neighbor 10.25.1.2 remote-as 65100
 neighbor 10.56.1.6 remote-as 65200
```

R6
```
router bgp 65200
 bgp confederation identifier 200
 bgp confederation peers 65100
 neighbor 10.46.1.4 remote-as 65100
 neighbor 10.56.1.5 remote-as 65200
 neighbor 10.67.1.7 remote-as 300
```

R7
```
router bgp 300
 neighbor 10.67.1.6 remote-as 200
```

Confederations (Cont.)

- Route reflectors can be used within the member AS like normal iBGP peerings.
- The BGP MED attribute is transitive to all other member ASs but does not leave the confederation.
- The LOCAL_PREF attribute is transitive to all other member ASs but does not leave the confederation.
- The next-hop address for external confederation routes does not change as the route is exchanged between member ASs.
- AS_CONFED_SEQUENCE is removed from AS_Path when the route is advertised outside the confederation.

Example 11-19 shows R1's BGP table, which displays all the routes advertised from this topology.

Notice that R2 removed the member AS ASNs from the route as it is advertised externally. AS 100 is not aware that AS 200 is a confederation.

Example 11-19 *AS 100's BGP Table*

```
R1-AS100# show bgp ipv4 unicast | begin Network
     Network          Next Hop            Metric LocPrf Weight Path
 *>  10.1.1.0/24      0.0.0.0                  0         32768 ?
 *>  10.7.7.0/24      10.12.1.2                              0 200 300 i
 *   10.12.1.0/24     10.12.1.2                0             0 200 ?
 *>                   0.0.0.0                  0         32768 ?
 *>  10.23.1.0/24     10.12.1.2                0             0 200 ?
 *>  10.25.1.0/24     10.12.1.2                0             0 200 ?
 *>  10.46.1.0/24     10.12.1.2                              0 200 ?
 *>  10.56.1.0/24     10.12.1.2                              0 200 ?
 *>  10.67.1.0/24     10.12.1.2                              0 200 ?
 *>  10.78.1.0/24     10.12.1.2                              0 200 300 ?
```


Confederations (Cont.)

Example 11-20 shows R2's BGP table, which participates in the member AS 65100. Notice that the next-hop IP address is not modified for the 10.7.7.0/24 prefix that was advertised by R7, even though it passed a different member AS. AS_CONFED_SEQUENCE is listed in parentheses to indicate that it passed through sub AS 65200 in the AS 200 confederation.

Example 11-20 *R2's BGP Table*

```
R2# show bgp ipv4 unicast | begin Network
     Network          Next Hop            Metric LocPrf Weight Path
 *>  10.1.1.0/24      10.12.1.1              111             0 100 ?
 *>  10.7.7.0/24      10.67.1.7                0    100      0 (65200) 300 i
 *>  10.12.1.0/24     0.0.0.0                  0         32768 ?
 *                    10.12.1.1              111             0 100 ?
 *>  10.23.1.0/24     0.0.0.0                  0         32768 ?
 *   10.25.1.0/24     10.25.1.5                0    100      0 (65200) ?
 *>                   0.0.0.0                  0         32768 ?
 *>  10.46.1.0/24     10.56.1.6                0    100      0 (65200) ?
 *>  10.56.1.0/24     10.25.1.5                0    100      0 (65200) ?
 *>  10.67.1.0/24     10.56.1.6                0    100      0 (65200) ?
 *>  10.78.1.0/24     10.67.1.7                0    100      0 (65200) 300 ?
Processed 8 prefixes, 10 paths
```

Confederations (Cont.)

Example 11-21 shows the full NRLI information from the perspective of R4 for the prefix 10.7.7.0/24 that was advertised from R7. Notice that the NRLI includes the fields *confedinternal* and *confed-external* based on whether the NLRI was received within the same member AS or a different one.

Example 11-21 *Confederation NLRI*

```
R4# show bgp ipv4 unicast 10.7.7.0/24
! Output omitted for brevity
BGP routing table entry for 10.7.7.0/24, version 504
Paths: (2 available, best #1, table default)
  Advertised to update-groups:
     3
  Refresh Epoch 1
  (65200) 300
    10.67.1.7 from 10.34.1.3 (192.168.3.3)
      Origin IGP, metric 0, localpref 100, valid, confed-internal, best
      Originator: 192.168.2.2, Cluster list: 192.168.3.3
      rx pathid: 0, tx pathid: 0x0
  Refresh Epoch 1
  (65200) 300
    10.67.1.7 from 10.46.1.6 (192.168.6.6)
      Origin IGP, metric 0, localpref 100, valid, confed-external
      rx pathid: 0, tx pathid: 0
```

Multiprotocol BGP for IPv6

- Multiprotocol BGP (MP-BGP) enables BGP to carry NLRI for multiple protocols, such as IPv4, IPv6, and MPLS Layer 3 Virtual Private Network L3VPN.

Multiprotocol BGP for IPv6
MP-BGP

MP-BGP RFC 4760 defines the following new features:

- New address family identifier (AFI) model
- New BGPv4 optional and nontransitive attributes:
- Multiprotocol reachable NLRI
- Multiprotocol unreachable NLRI

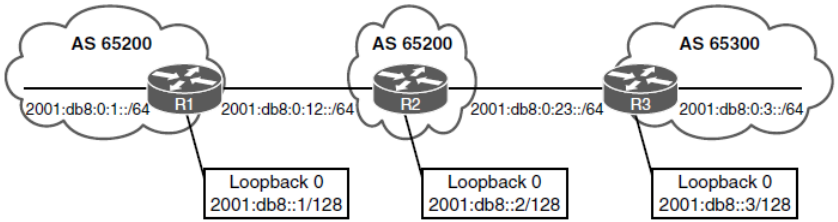

Figure 11-20 *IPv6 Sample Topology*

The multiprotocol reachable NLRI attribute describes IPv6 route information, and the multiprotocol unreachable NLRI attribute withdraws the IPv6 route from service. The attributes are optional and nontransitive, so if an older router does not understand the attributes, the information can just be ignored.

During the open message negotiation, the BGP peer routers exchange capabilities. The MP-BGP extensions include an AFI that describes the supported protocols, along with SAFI attribute fields that describe whether the prefix applies to the unicast or multicast routing table:

- IPv4 unicast: AFI:1, SAFI:1
- IPv6 unicast: AFI:2, SAFI:1

Figure 11-20 shows a simple topology with three different ASs and R2 forming an eBGP session with R1 and R3. All of R1's links are configured to FE80::1, all of R2's links are set to FE80::2, and all of R3's links are configured to FE80::3. This topology is used throughout this section.

IPv6 Configuration

- BGP configuration rules that apply to IPv4 apply to IPv6, except that the IPv6 address family must be initialized, and the neighbor is activated. Routers with only IPv6 addressing must statically define the BGP RID to allow sessions to form.
- The TCP session used by BGP is a Layer 4 protocol (TCP port 179), and it can use either an IPv4 or IPv6 address to form a session adjacency and exchange routes.
- Unique global unicast addressing is the recommended method for BGP peering to avoid operational complexity. BGP peering using the link-local address may introduce risk if the address is not manually assigned to an interface.
- A hardware failure or cable move changes the MAC address, resulting in a new link-local address. This causes the session to fail because the stateless address autoconfiguration generates a new IP address.

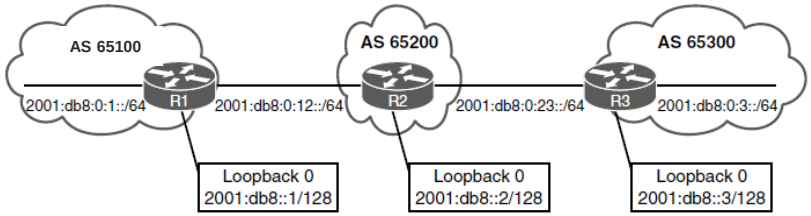

Figure 11-20 *IPv6 Sample Topology*

IPv6 Configuration (Cont.)

Example 11-22 shows the IPv6 BGP configuration for R1, R2, and R3. The peering uses global unicast addressing for establishing the session. The BGP RID has been set to the IPv4 loopback format used throughout this book. R1 advertises all its networks through redistribution, and R2 and R3 use the network statement to advertise all their connected networks.

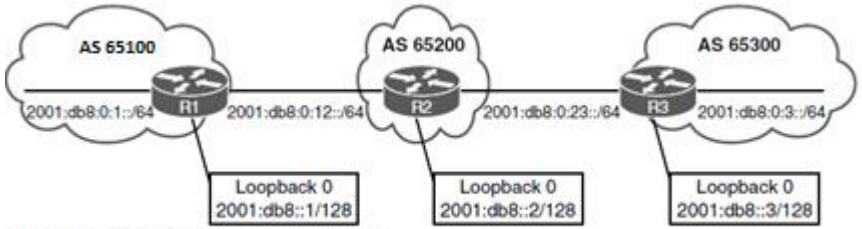

Figure 11-20 *IPv6 Sample Topology*

R1

```
router bgp 65100
 bgp router-id 192.168.1.1
 bgp log-neighbor-changes
 no bgp default ipv4-unicast
 neighbor 2001:DB8:0:12::2 remote-as 65200
 !
address-family ipv6
  neighbor 2001:DB8:0:12::2 activate
  redistribute connected
```

R2

```
router bgp 65200
 bgp router-id 192.168.2.2
 bgp log-neighbor-changes
 no bgp default ipv4-unicast
 neighbor 2001:DB8:0:12::1 remote-as 65100
 neighbor 2001:DB8:0:23::3 remote-as 65300
!
 address-family ipv6
  neighbor 2001:DB8:0:12::1 activate
  neighbor 2001:DB8:0:23::3 activate
  network 2001:DB8::2/128
  network 2001:DB8:0:12::/64
  network 2001:DB8:0:23::/64
```

R3

```
router bgp 65300
 bgp router-id 192.168.3.3
 bgp log-neighbor-changes
 no bgp default ipv4-unicast
 neighbor 2001:DB8:0:23::2 remote-as 65200
 !
 address-family ipv6
  neighbor 2001:DB8:0:23::2 activate
  network 2001:DB8::3/128
  network 2001:DB8:0:3::/64
  network 2001:DB8:0:23::/64
```

IPv6 Configuration (Cont.)

IPv4 unicast routing capability is advertised by default in IOS unless the neighbor is specifically shut down within the IPv4 address family or globally within the BGP process with the **no bgp default ipv4-unicast** command.

Routers exchange AFI capabilities during the initial BGP session negotiation. The command **show bgp ipv6 unicast neighbors** *ip-address* **[detail]** displays detailed information about whether the IPv6 capabilities were negotiated successfully.

Example 11-23 shows the fields that should be examined for IPv6 session establishment and route advertisement.

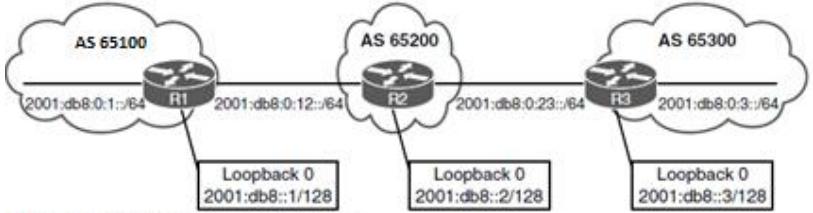

Figure 11-20 *IPv6 Sample Topology*

Example 11-23 *Viewing BGP Neighbors for IPv6 Capabilities*

```
R1# show bgp ipv6 unicast neighbors 2001:DB8:0:12::2
! Output omitted for brevity
BGP neighbor is 2001:DB8:0:12::2,  remote AS 65200, external link
  BGP version 4, remote router ID 192.168.2.2
  BGP state = Established, up for 00:28:25
  Last read 00:00:54, last write 00:00:34, hold time is 180, keepalive interval is
60 seconds
  Neighbor sessions:
    1 active, is not multisession capable (disabled)
  Neighbor capabilities:
    Route refresh: advertised and received(new)
    Four-octets ASN Capability: advertised and received
    Address family IPv6 Unicast: advertised and received
    Enhanced Refresh Capability: advertised and received
 ..
 For address family: IPv6 Unicast
  Session: 2001:DB8:0:12::2
  BGP table version 13, neighbor version 13/0
  Output queue size : 0
  Index 1, Advertise bit 0
  1 update-group member
  Slow-peer detection is disabled
  Slow-peer split-update-group dynamic is disabled
                                  Sent       Rcvd
  Prefix activity:                ----       ----
    Prefixes Current:                3          5 (Consumes 520 bytes)
    Prefixes Total:                  6         10
```

IPv6 Configuration (Cont.)

The **show bgp ipv6 unicast summary** command displays a status summary of the sessions, including the number of routes that have been exchanged and the session uptime.

Example 11-24 highlights the IPv6 AFI neighbor status for R2. Notice that the two neighbor adjacencies have been up for about 25 minutes. Neighbor 2001:db8:0:12::1 is advertising three routes, and neighbor 2001:db8:0:23::3 is advertising three routes.

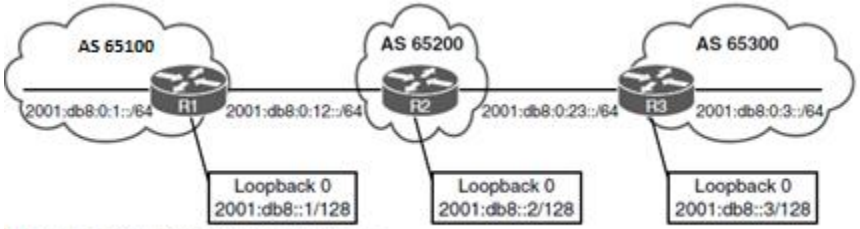

Figure 11-20 *IPv6 Sample Topology*

Example 11-24 *Verification of IPv6 BGP Session*

```
R2# show bgp ipv6 unicast summary
BGP router identifier 192.168.2.2, local AS number 65200
BGP table version is 19, main routing table version 19
7 network entries using 1176 bytes of memory
8 path entries using 832 bytes of memory
3/3 BGP path/bestpath attribute entries using 456 bytes of memory
2 BGP AS-PATH entries using 48 bytes of memory
0 BGP route-map cache entries using 0 bytes of memory
0 BGP filter-list cache entries using 0 bytes of memory
BGP using 2512 total bytes of memory
BGP activity 7/0 prefixes, 8/0 paths, scan interval 60 secs

Neighbor        V    AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down  State/PfxRcd
2001:DB8:0:12::1 4 65100      35      37       19    0    0 00:25:08        3
2001:DB8:0:23::3 4 65300      32      37       19    0    0 00:25:11        3
```

IPv6 Configuration (Cont.)

Example 11-25 shows the IPv6 unicast BGP table for R1, R2, and R3. Notice that some of the routes include the unspecified address (::) as the next hop. The unspecified address indicates that the local router is generating the prefix for the BGP table. The weight value 32,768 also indicates that the prefix is locally originated by the router.

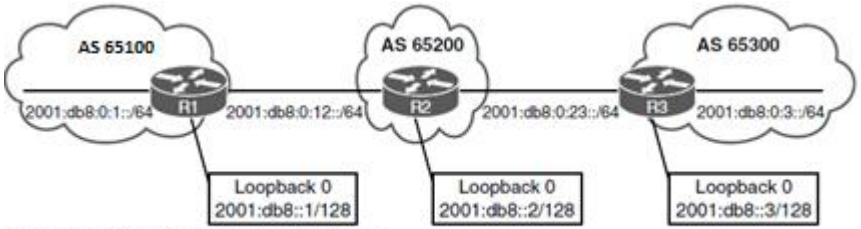

Figure 11-20 *IPv6 Sample Topology*

Example 11-25 *Viewing the IPv6 BGP Table*

```
R1# show bgp ipv6 unicast
BGP table version is 13, local router ID is 192.168.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 *>  2001:DB8::1/128  ::                        0         32768 ?
 *>  2001:DB8::2/128  2001:DB8:0:12::2         0             0 65200 i
 *>  2001:DB8::3/128  2001:DB8:0:12::2                       0 65200 65300 i
 *>  2001:DB8:0:1::/64 ::                       0         32768 ?
 *>  2001:DB8:0:3::/64 2001:DB8:0:12::2                      0 65200 65300 i
 *   2001:DB8:0:12::/64 2001:DB8:0:12::2       0             0 65200 i
 *>                    ::                       0         32768 ?
 *>  2001:DB8:0:23::/64 2001:DB8:0:12::2                     0 65200 65300 i
```

```
R2# show bgp ipv6 unicast | begin Network
     Network          Next Hop            Metric LocPrf Weight Path
 *>  2001:DB8::1/128  2001:DB8:0:12::1       0             0 65100 ?
 *>  2001:DB8::2/128  ::                       0         32768 i
 *>  2001:DB8::3/128  2001:DB8:0:23::3       0             0 65300 i
 *>  2001:DB8:0:1::/64 2001:DB8:0:12::1       0             0 65100 ?
 *>  2001:DB8:0:3::/64 2001:DB8:0:23::3       0             0 65300 i
 *>  2001:DB8:0:12::/64 ::                    0         32768 i
 *                    2001:DB8:0:12::1       0             0 65100 ?
 *>  2001:DB8:0:23::/64 ::                      0          32768 i
                      2001:DB8:0:23::3       0             0 65300 i
```

```
R3# show bgp ipv6 unicast | begin Network
     Network          Next Hop            Metric LocPrf Weight Path
 *>  2001:DB8::1/128  2001:DB8:0:23::2                     0 65200 65100 ?
 *>  2001:DB8::2/128  2001:DB8:0:23::2       0             0 65200 i
 *>  2001:DB8::3/128  ::                       0         32768 i
 *>  2001:DB8:0:1::/64 2001:DB8:0:23::2                     0 65200 65100 ?
 *>  2001:DB8:0:3::/64 ::                     0         32768 i
 *>  2001:DB8:0:12::/64 2001:DB8:0:23::2      0             0 65200 i
 *>  2001:DB8:0:23::/64 ::                    0         32768 i
```

IPv6 Configuration (Cont.)

You can view the BGP path attributes for an IPv6 route by using the **show bgp ipv6 unicast** *prefix*/*prefix-length* command.

Example 11-26 shows R3 examining R1's loopback address. Some of the common fields, such as those for AS_Path, origin, and local preference, are identical to those for IPv4 routes.

Example 11-26 *Viewing the BGP Path Attributes for an IPv6 Route*

```
R3# show bgp ipv6 unicast 2001:DB8::1/128
BGP routing table entry for 2001:DB8::1/128, version 9
Paths: (1 available, best #1, table default)
  Not advertised to any peer
  Refresh Epoch 2
  65200 65100
    2001:DB8:0:23::2 (FE80::2) from 2001:DB8:0:23::2 (192.168.2.2)
      Origin incomplete, localpref 100, valid, external, best
      rx pathid: 0, tx pathid: 0x0
```

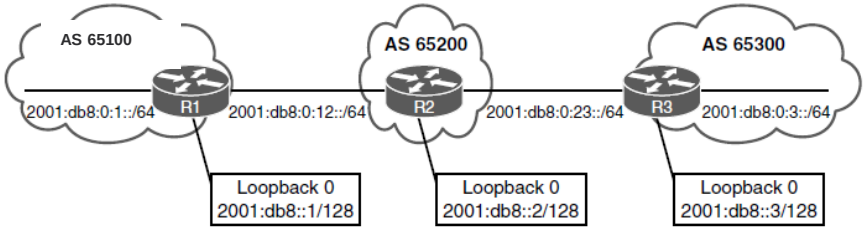

Figure 11-20 *IPv6 Sample Topology*

IPv6 Summarization

The same process for summarizing or aggregating IPv4 routes occurs with IPv6 routes, and the format is identical except that the configuration is placed under the IPv6 address family using the **aggregate-address** *prefix/prefix-length* **[summary-only] [as-set]** command.

Let's revisit the previous IPv6 deployment in Figure 11-20, but now summarize all the loopback addresses 2001:db8:0:1/128, 2001:db8:0:2/128, and 2001:db8:0:3/128) along with the peering link between R1 and R2 2001:db8:0:12/64) on R2. The configuration would look as shown in Example 11-28.

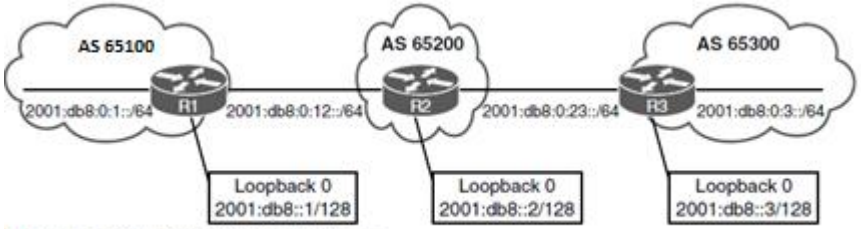

Figure 11-20 *IPv6 Sample Topology*

Example 11-28 *Configuring IPv6 BGP Aggregation on R2*

```
router bgp 65200
 bgp router-id 192.168.2.2
 bgp log-neighbor-changes
 neighbor 2001:DB8:0:12::1 remote-as 65100
 neighbor 2001:DB8:0:23::3 remote-as 65300
 !
 address-family ipv4
  no neighbor 2001:DB8:0:12::1 activate
  no neighbor 2001:DB8:0:23::3 activate
 exit-address-family
 !
 address-family ipv6
 bgp scan-time 6
  network 2001:DB8::2/128
  network 2001:DB8:0:12::/64
  aggregate-address 2001:DB8::/59 summary-only
  neighbor 2001:DB8:0:12::1 activate
  neighbor 2001:DB8:0:23::3 activate
 exit-address-family
```

IPv6 Summarization (Cont.)

Example 11-29 shows the BGP tables on R1 and R3. All of the smaller routes are aggregated and suppressed into 2001:db8::/59 as expected.

The summarization of the IPv6 loopback addresses (2001:db8:0:1/128, 2001:db8:0:2/128, and 2001:db8:0:3/128) is fairly simple as they all fall into the base IPv6 summary range 2001:db8:0:0::/64. The fourth hextet, beginning with a decimal value of 1, 2, or 3, would consume only 2 bits; the range could be summarized easily into the 2001:db8:0:0::/62 (or 2001:db8::/62) network range.

The peering link between R1 and R2 2001:db8:0:12::/64) requires thinking in hex first rather than in decimal values. The fourth hextet carries a decimal value of 18 (not 12), which requires 5 bits minimum. Table 11-6 lists the bits needed for summarization, the IPv6 summary address, and the component networks in the summary range.

Example 11-29 *Verification of IPv6 Route Aggregation*

```
R3# show bgp ipv6 unicast | b Network
     Network            Next Hop            Metric LocPrf Weight Path
 *>  2001:DB8::/59      2001:DB8:0:23::2         0             0 65200 i
 *>  2001:DB8::3/128    ::                       0         32768 i
 *>  2001:DB8:0:3::/64  ::                       0         32768 i
 *>  2001:DB8:0:23::/64 ::                       0         32768 i

R1# show bgp ipv6 unicast | b Network
     Network            Next Hop            Metric LocPrf Weight Path
 *>  2001:DB8::/59      2001:DB8:0:12::2         0             0 65200 i
 *>  2001:DB8::1/128    ::                       0         32768 ?
 *>  2001:DB8:0:1::/64  ::                       0         32768 ?
 *>  2001:DB8:0:12::/64 ::                       0         32768 ?
 *>  2001:DB8:0:23::/64 2001:DB8:0:12::2        0 65200 65300 i
```

Table 11-6 IPv6 Summarization Table

Bits Needed	Summary Address	Component Networks
2	2001:db8:0:0::/62	2001:db8:0:0::/64 through 2001:db8:0:3::/64
3	2001:db8:0:0::/61	2001:db8:0:0::/64 through 2001:db8:0:7::/64
4	2001:db8:0:0::/60	2001:db8:0:0::/64 through 2001:db8:0:F::/64
5	2001:db8:0:0::/59	2001:db8:0:0::/64 through 2001:db8:0:1F::/64
6	2001:db8:0:0::/58	2001:db8:0:0::/64 through 2001:db8:0:3F::/64

IPv6 Summarization (Cont.)

Currently the peering link between R2 and R3 (2001:db8:0:23::/64) is not being summarized and suppressed, as it is still visible in R1's routing table in Example 11-35. The hex value of 23, which is typically written as 0x23, converts to decimal value 35; which requires 6 bits. The summarized network range must be changed to 2001:db8::/58 for summarization of the 2001:db9:0:23::/64 network to occur. Example 11-30 shows the configuration change being made to R2.

Example 11-31 verifies that the 2001:db8:0:23::/64 is now within the aggregate address space and is no longer being advertised to R1.

Example 11-30 *Configuration Change to Summarize the 2001:db8:0:23::/64 Network*

```
R2# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
R2(config)# router bgp 65200
R2(config-router)# address-family ipv6 unicast
R2(config-router-af)# no aggregate-address 2001:DB8::/59 summary-only
R2(config-router-af)# aggregate-address 2001:DB8::/58 summary-only
```

Example 11-31 *Verification of Summarization of the 2001:db8:0:23::/64 Network*

```
R1# show bgp ipv6 unicast | b Network
     Network            Next Hop            Metric LocPrf Weight Path
 *>  2001:DB8::/58      2001:DB8:0:12::2         0             0 65200 i
 *>  2001:DB8::1/128    ::                       0         32768 ?
 *>  2001:DB8:0:1::/64  ::                       0         32768 ?
 *>  2001:DB8:0:12::/64 ::                       0         32768 ?
```

Multiprotocol BGP for IPv6
IPv6 over IPv4

BGP can exchange routes using either an IPv4 or IPv6 TCP session. In a typical deployment, IPv4 routes are exchanged using a dedicated IPv4 session, and IPv6 routes are exchanged with a dedicated IPv6 session. However, it is possible to share IPv6 routes over an IPv4 TCP session or IPv4 routes over an IPv6 TCP session, and it is possible to share IPv4 and IPv6 routes using a single BGP session.

Example 11-32 shows how to configure the exchange of IPv6 routes over IPv4 using the topology shown in Figure 11-20. Notice that the IPv6 neighbors must be activated, and the routers are injected into BGP under the IPv6 address family.

Example 11-32 *Configuring IPv6 Route Exchange over an IPv4 BGP Session*

R1

```
router bgp 65100
 bgp router-id 192.168.1.1
 no bgp default ipv4-unicast
 neighbor 10.12.1.2 remote-as 65200
 !
address-family ipv6 unicast
  redistribute connected
  neighbor 10.12.1.2 activate
```

R2

```
router bgp 65200
 bgp router-id 192.168.2.2
 no bgp default ipv4-unicast
 neighbor 10.12.1.1 remote-as 65100
 neighbor 10.23.1.3 remote-as 65300
 !
 address-family ipv6 unicast
  bgp scan-time 6
  network 2001:DB8::2/128
  network 2001:DB8:0:12::/64
  aggregate-address 2001:DB8::/62 summary-only
  neighbor 10.12.1.1 activate
  neighbor 10.23.1.3 activate
```

R3

```
router bgp 65300
 bgp router-id 192.168.3.3
 no bgp default ipv4-unicast
 neighbor 10.23.1.2 remote-as 65200
 !
 address-family ipv6 unicast
  network 2001:DB8::3/128
  network 2001:DB8:0:3::/64
  network 2001:DB8:0:23::/64
  neighbor 10.23.1.2 activate
```

IPv6 over IPv4 (Cont.)

Example 11-34 shows the IPv6 BGP table for all three routers, which verifies that the routes have been successfully advertised.

The IPv6 routes advertised over an IPv4 BGP session are assigned an IPv4-mapped IPv6 address in the format (::FFFF:*xx.xx.xx.xx*) for the next hop, where *xx.xx.xx.xx* is the IPv4 address of the BGP peering. This is not a valid forwarding address, so the IPv6 route does not populate the RIB.

Example 11-35 shows a quick connectivity test between R1 and R3 and confirms that connectivity cannot be maintained.

Example 11-35 *Checking Connectivity Between R1 and R3*

```
R1# ping 2001:DB8:0:3::3
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:DB8:0:3::3, timeout is 2 seconds:

% No valid route for destination
Success rate is 0 percent (0/1)
```

Example 11-34 *Viewing IPv6 Routes Exchanged over an IPv4 BGP Session*

```
R1# show bgp ipv6 unicast | begin Network
     Network          Next Hop            Metric LocPrf Weight Path
 *   2001:DB8::/62    ::FFFF:10.12.1.2         0             0 65200 i
 *>  2001:DB8::1/128  ::                       0         32768 ?
 *>  2001:DB8:0:1::/64 ::                      0         32768 ?
 *   2001:DB8:0:12::/64 ::FFFF:10.12.1.2      0             0 65200 i
 *>                   ::                      0         32768 ?

R2# show bgp ipv6 unicast | begin Network
     Network          Next Hop            Metric LocPrf Weight Path
 *>  2001:DB8::/62    ::                                32768 i
 S   2001:DB8::1/128  ::FFFF:10.12.1.1    0             0 65100 ?
 s>  2001:DB8::2/128  ::                  0         32768 i
 s   2001:DB8::3/128  ::FFFF:10.23.1.3    0             0 65300 i
 s   2001:DB8:0:1::/64 ::FFFF:10.12.1.1   0             0 65100 ?
 s   2001:DB8:0:3::/64 ::FFFF:10.23.1.3   0             0 65300 i
 *   2001:DB8:0:12::/64 ::FFFF:10.12.1.1  0             0 65100 ?
 *>                   ::                  0         32768 i
 *   2001:DB8:0:23::/64 ::FFFF:10.23.1.3  0             0 65300 i

R3# show bgp ipv6 unicast | begin Network
     Network          Next Hop            Metric LocPrf Weight Path
 *   2001:DB8::/62    ::FFFF:10.23.1.2         0             0 65200 i
 *>  2001:DB8::3/128  ::                       0         32768 i
 *>  2001:DB8:0:3::/64 ::                      0         32768 i
 *   2001:DB8:0:12::/64 ::FFFF:10.23.1.2      0             0 65200 i
 *>  2001:DB8:0:23::/64 ::                     0         32768 i
```

IPv6 over IPv4 (Cont.)

To correct the problem, the BGP route map needs to manually set the IPv6 next hop.

Example 11-36 shows the BGP configuration for R1, R2, and R3.

Example 11-36 *Route Map to Manually Set the IPv6 Next Hop*

R1

```
route-map FromR1R2Link permit 10
 set ipv6 next-hop 2001:DB8:0:12::1
!
router bgp 65100
 address-family ipv6 unicast
  neighbor 10.12.1.2 route-map FromR1R2LINK out
```

R2

```
route-map FromR2R1LINK permit 10
 set ipv6 next-hop 2001:DB8:0:12::2
route-map FromR2R3LINK permit 10
 set ipv6 next-hop 2001:DB8:0:23::2
!
router bgp 65200
 address-family ipv6 unicast
  neighbor 10.12.1.1 route-map FromR2R1LINK out
  neighbor 10.23.1.3 route-map FromR2R3LINK out
```

R3

```
route-map FromR3R2Link permit 10
 set ipv6 next-hop 2001:DB8:0:23::3
!
router bgp 65300
 address-family ipv6 unicast
  neighbor 10.23.1.2 route-map FromR3R2Link out
```

IPv6 over IPv4 (Cont.)

Example 11-37 shows the BGP table after the IPv6 next-hop address is manually set on the outbound route map. The next-hop IP address is valid, and the route can now be installed into the RIB.

Example 11-37 *Viewing IPv6 Routes After Manually Setting the IPv6 Next Hop*

```
R1# show bgp ipv6 unicast | begin Network
   Network            Next Hop            Metric LocPrf Weight Path
*> 2001:DB8::/62      2001:DB8:0:12::2         0             0 65200 i
*> 2001:DB8::1/128    ::                       0         32768 ?
*> 2001:DB8:0:1::/64  ::                       0         32768 ?
*> 2001:DB8:0:12::/64 ::                       0         32768 ?
*                     2001:DB8:0:12::2         0             0 65200 i
```

```
R2# show bgp ipv6 unicast | begin Network
    Network            Next Hop            Metric LocPrf Weight Path
*> 2001:DB8::/62      ::                                   32768 i
s> 2001:DB8::1/128    2001:DB8:0:12::1         0             0 65100 ?
s> 2001:DB8::2/128    ::                       0         32768 i
s> 2001:DB8::3/128    2001:DB8:0:23::3         0             0 65300 i
s> 2001:DB8:0:1::/64  2001:DB8:0:12::1         0             0 65100 ?
s> 2001:DB8:0:3::/64  2001:DB8:0:23::3         0             0 65300 i
*> 2001:DB8:0:12::/64 ::                       0         32768 i
r> 2001:DB8:0:23::/64 2001:DB8:0:23::3         0             0 65300 i
```

```
R3# show bgp ipv6 unicast | begin Network
   Network            Next Hop            Metric LocPrf Weight Path
*> 2001:DB8::/62      2001:DB8:0:23::2
                                               0             0 65200 i
*> 2001:DB8::3/128    ::                       0         32768 i
*> 2001:DB8:0:3::/64  ::                       0         32768 i
*> 2001:DB8:0:12::/64 2001:DB8:0:23::2         0             0 65200 i
*> 2001:DB8:0:23::/64 ::                       0         32768 i
```

Prepare for the Exam

Prepare for the Exam

Key Topics for Chapter 11

Description	Description
Autonomous system numbers (ASNs)	BGP tables
Path attributes	BGP prefix validity check and installation
Inter-router communication	BGP database processing
BGP single and multi-hop sessions	Viewing all the paths and path attributes for a specific router
BGP messages	Viewing the Adj-RIB-Out for a specific BGP peer
BGP neighbor states	Understanding BGP session types and behaviors
Basic BGP configuration	iBGP full mesh requirement
Verification of BGP sessions	Peering using loopback addresses
BGP summary fields	eBGP peerings

Prepare for the Exam

Key Topics for Chapter 11 (Cont.)

Description
eBGP and iBGP topologies
Next-hop manipulation
Route reflectors
Route reflector rules
Confederations
IPv6 BGP configuration
IPv6 summarization
Advertising IPv6 prefixes over an IPv4 BGP session

Prepare for the Exam

Key Terms for Chapter 11

Term	Term
address family	optional non-transitive
AS_Path	path vector routing protocol
autonomous system	route reflector
BGP confederation	route reflector client
eBGP session	well-known mandatory
iBGP session	well-known discretionary
optional transitive	Loc-RIB table

Command Reference for Chapter 11

Task	Command Syntax
Initialize the BGP router process	**router bgp** *as-number*
Statically configure the BGP router ID	**bgp router-id** *router-id*
Identify a BGP peer to establish a session with	**neighbor** *ip-address* **remote-as** *as-number*
Configure the BGP session timers	**neighbor** *ip-address* **timers** *keepalive holdtime [minimum-holdtime]*
Specify the source interface for BGP packets for a specific BGP peer	**neighbor** *ip-address* **update-source** *interface-id*
Specify the ASN at which the BGP confederation should appear	**bgp confederation identifier** *as-number*
Specify any BGP confederation member ASs that this router will peer with	**bgp confederation peers** *member-asn*

Command Reference for Chapter 11 (Cont.)

Task	Command Syntax
Disable the automatic IPv4 address family configuration mode	**no bgp default ipv4-unicast**
Initialize a specific address family and sub-address family	**address-family** *afi safi*
Activate a BGP neighbor for a specific address family	**neighbor** *ip-address* **activate**
Advertise a network into BGP	**network** *network* **mask** *subnet-mask* *[**route-map** route-map-name]*
Modify the next-hop IP address on a prefix advertisements to match that of the IP address used for the BGP session	**neighbor** *ip-address* **next-hop-self [all]**
Configure the associated BGP peer as a router reflector client	**neighbor** *ip-address* **route-reflector-client**

Command Reference for Chapter 11 (Cont.)

Task	Command Syntax
Display the contents of the BGP database	**show bgp** *afi safi* [network] [detailed]
Display a summary of the BGP tables and neighbor peering sessions	**show bgp** *afi safi* **summary**
Display the negotiated the BGP settings with a specific peer and the number prefixes exchanged with that peer	**show bgp** *afi safi* **neighbors** ip-address
Display the Adj-RIB-out BGP table for a specific BGP neighbor	**show bgp** *afi safi* **neighbor** *ip-address* **advertised routes**


CISCO