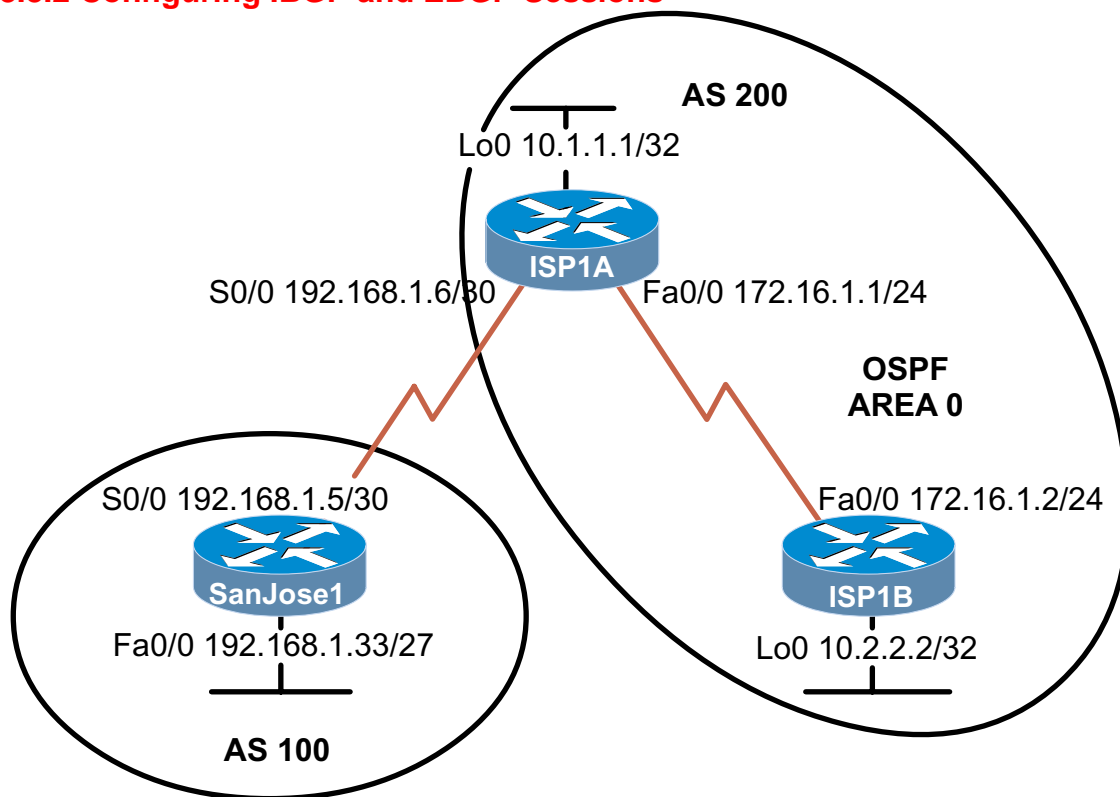


8.8.2 Configuring IBGP and EBGp Sessions



Objective

In this lab, you configure both IBGP and EBGp. In order for IBGP peers to correctly exchange routing information, you must also use the **next-hop-self** and **aggregate-address** commands.

Scenario

The International Travel Agency runs BGP on its SanJose1 router externally with ISP1A (AS 200). ISP1A also runs BGP internally between two of its routers, ISP1A and ISP1B. Your job is to configure both EBGp and IBGP for this internetwork.

Step 1

Build and configure the network according to the diagram, but do not configure a routing protocol. Configure a loopback interface on the ISP1A's and ISP1B's routers as shown. These loopbacks will be used with BGP **neighbor** statements for increased fault tolerance.

Use **ping** to test connectivity between the directly connected routers.

Note: The SanJose1 router will not be able to reach the ISP1B router.

Step 2

Configure OSPF between the ISP1A and ISP1B routers with the same commands:

```
(config)#router ospf 1
(config-router)#network 10.0.0.0 0.255.255.255 area 0
(config-router)#network 172.16.1.0 0.0.0.255 area 0
```

The **network 10.0.0.0** statement is necessary so that the routers will exchange information about their loopback interfaces. BGP needs this information in order to establish a TCP connection.

Step 3

Configure IBGP between the ISP1A and ISP1B routers. On the ISP1A router, enter the following configuration:

```
(config)#router bgp 200
(config-router)#no auto-summary
(config-router)#neighbor 10.2.2.2 remote-as 200
(config-router)#neighbor 10.2.2.2 update-source lo0
```

This topology uses VLSM, so automatic summarization (along classful boundaries) should be disabled with the **no auto-summary** command.

As for the other BGP commands, notice that the neighbor, 10.2.2.2 (ISP1B's loopback address), is not in a *remote* AS at all. Both routers are in the same AS (AS 200). Even so, there is no "local AS" command, so you still use the **remote-as** keyword to specify ISP1B's AS membership.

If multiple pathways to the neighbor exist, then the router can use any IP interface to speak BGP with that neighbor. The **update-source lo0** command instructs the router to use interface loopback 0 for TCP connections. This command is typically used in all IBGP configurations. Without this command, BGP routers can use only the closest IP interface to the peer. The capability to use any operational interface provides BGP with robustness in case the link to the closet interface fails.

Because BGP will eventually advertise outside networks that are not part of the OSPF area, you must also enter the following command on the ISP1A router:

```
ISP1A(config)#router bgp 200
ISP1A(config-router)#no synchronization
```

The **no synchronization** command permits BGP to advertise networks without caring whether the IGP (in this case, OSPF) has the route. Usually, a BGP speaker does not advertise a route to an external neighbor unless that route is local or exists in the IGP.

Step 4

Complete the IBGP configuration by entering the following commands on the peer router, ISP1B:

```
ISP1B(config)#router bgp 200
ISP1B(config-router)#no auto-summary
ISP1B(config-router)#no synchronization
ISP1B(config-router)#neighbor 10.1.1.1 remote-as 200
ISP1B(config-router)#neighbor 10.1.1.1 update-source lo0
```

Verify that ISP1A and ISP1B have become BGP neighbors by issuing the **show ip bgp neighbors** command on ISP1B (see the following partial output). If the BGP state is not established, troubleshoot your connection.

1. According to the output of this command, is the link between ISP1B and ISP1A internal or external?

```
ISP1B#show ip bgp neighbors
BGP neighbor is 10.1.1.1, remote AS 200, internal link
  BGP version 4, remote router ID 10.1.1.1
    BGP state = Established, up for 00:02:18
```

Step 5

Configure ISP1A to run EBGp with SanJose1. Enter the following commands on ISP1A:

```
ISP1A(config)#router bgp 200
ISP1A(config-router)#neighbor 192.168.1.5 remote-as 100
ISP1A(config-router)#network 172.16.1.0 mask 255.255.255.0
```

Because EBGp sessions are almost always established over point-to-point links, there's no reason to use the **update-source** keyword in this configuration. Only one path exists between the peers; if it goes down, alternative paths are not available.

Note: The **mask** keyword can be applied to tell BGP to advertise a particular subnet when autosummarization is disabled.

Step 6

Finally, configure SanJose1 as an EBGp peer to ISP1A:

```
SanJose1(config)#router bgp 100
SanJose1(config-router)#no auto-summary
SanJose1(config-router)#neighbor 192.168.1.6 remote-as 200
SanJose1(config-router)#network 192.168.1.32 mask 255.255.255.224
```

Use the **show ip bgp neighbors** command to verify that SanJose1 and ISP1A have reached the Established state. Troubleshoot, if necessary.

Step 7

Observe the BGP neighbor negotiation process. On SanJose1, shut down the serial interface connected to ISP1A:

```
SanJose1(config)#interface serial 0/0
SanJose1(config-if)#shutdown
```

After you shut down the interface, issue the **debug ip bgp** command at SanJose1's console:

```
SanJose1#debug ip bgp
```

Bring SanJose1's serial interface back up, and observe the output of the **debug** command (a partial output is shown here):

```
01:09:42: BGP: 192.168.1.6 went from Idle to Active
01:09:42: BGP: 192.168.1.6 open active, delay 17576ms
01:09:59: BGP: 192.168.1.6 open active, local address 10.0.0.2
01:09:59: BGP: 192.168.1.6 went from Active to OpenSent
01:09:59: BGP: 192.168.1.6 sending OPEN, version 4, my as: 100
01:09:59: BGP: 192.168.1.6 send message type 1, length
    (incl. header) 41
01:09:59: BGP: 192.168.1.6 rcv message type 1, length
    (excl. header) 22
01:09:59: BGP: 192.168.1.6 rcv OPEN, version 4
01:09:59: BGP: 192.168.1.6 rcv OPEN w/ OPTION parameter len: 12
01:09:59: BGP: 192.168.1.6 rcvd OPEN w/ optional parameter type 2
    (Capability) len 6
01:09:59: BGP: 192.168.1.6 OPEN has CAPABILITY code: 1, length 4
01:09:59: BGP: 192.168.1.6 OPEN has MP_EXT CAP for afi/safi: 1/1
01:09:59: BGP: 192.168.1.6 rcvd OPEN w/ optional parameter type 2
    (Capability) len 2
01:09:59: BGP: 192.168.1.6 OPEN has CAPABILITY code: 128, length 0
01:09:59: BGP: 192.168.1.6 went from OpenSent to OpenConfirm
01:10:00: BGP: 192.168.1.6 send message type 4, length
    (incl. header) 19
01:10:00: BGP: 192.168.1.6 rcv message type 4, length
    (excl. header) 0
01:10:00: BGP: 192.168.1.6 went from OpenConfirm to Established
01:10:00: BGP: 192.168.1.6 send message type 4, length
    (incl. header) 19
01:10:00: BGP: 192.168.1.6 rcv message type 4, length
    (excl. header) 0
```

1. Based on this output, which state followed Active?

2. Which state followed OpenConfirm?

Turn off debug with the command **undebg all**.

Step 8

Verify that ISP1A can ping SanJose1's FastEthernet address (192.168.1.33). These pings should be successful, but troubleshoot if necessary. Use **show ip route** to check SanJose1's routing table. SanJose1 should have a route to 172.16.0.0. Verify that SanJose1 can ping ISP1A's FastEthernet interface (172.16.1.1). This ping should also be successful.

While still at SanJose1, try pinging ISP1B's FastEthernet 0/0 (172.16.1.2), which is on the same network. This ping should not be successful. Check ISP1B's routing table, as shown here:

```
172.16.0.0/24 is subnetted, 1 subnets
C    172.16.1.0 is directly connected, FastEthernet0/0
10.0.0.0/32 is subnetted, 2 subnets
C    10.2.2.2 is directly connected, Loopback0
10.1.1.1 [110/2] via 172.16.1.1, 00:18:23, FastEthernet0/0
```

1. Can you tell why SanJose1 did not get a ping reply?

ISP1B does not have a route to 192.168.1.0. Because both of SanJose1's interfaces live on subnets within that network, ISP1B cannot respond to SanJose1's ping requests.

Check to see if ISP1B is receiving any route information about 192.168.1.0 via BGP. Issue the command **show ip bgp** at ISP1B's console.

```
ISP1B#show ip bgp
BGP table version is 3, local router ID is 10.2.2.2
Status codes: s suppressed, d damped, h history, * valid, > best,
               i - internal
Origin codes: i - IGP, e - EGP, ? - incomplete

Network          Next Hop        Metric LocPrf Weight Path
*>i172.16.1.0/24   10.1.1.1         0      100     0      i
* i192.168.1.32/27 192.168.1.5       0      100     0     200 i
```

2. A route to 192.168.1.32 exists in ISP1B's BGP table. According to the table, what is the next-hop IP address to 192.168.1.32 /27?
-

3. Is this IP address directly reachable as a next hop?
-

Recall that the BGP routers do not increment the next-hop address to their IBGP peers. In order for ISP1B to use ISP1A as the next hop, you must issue additional commands on ISP1A:

```
ISP1A(config)#router bgp 200
ISP1A(config-router)#neighbor 10.2.2.2 next-hop-self
```

After you issue these commands, reset BGP operation on ISP1A by entering the command **clear ip bgp ***.

Wait several seconds and then check ISP1B's BGP table with the **show ip bgp** command.

4. What is the next hop to the 192.168.1.32 /27 network now?
-

The 192.168.1.32 /27 network should now be in ISP1B's routing table.

5. What is the administrative distance for this route?
-

ISP1B should not be able to ping SanJose1's FastEthernet 0/0 (192.168.1.33).

Step 9

As a final connectivity test, return to SanJose1 and ping ISP1B's FastEthernet 0/0 (172.16.1.2). This ping should *not* be successful.

1. From what address is this ping sourced?

Check ISP1B's routing table again:

```

      172.16.0.0/24 is subnetted, 1 subnets
C       172.16.1.0 is directly connected, FastEthernet0/0
      10.0.0.0/32 is subnetted, 2 subnets
C       10.2.2.2 is directly connected, Loopback0
O       10.1.1.1 [110/2] via 172.16.1.1, 00:28:38, FastEthernet0/0
      192.168.1.0/27 is subnetted, 1 subnets
B       192.168.1.32 [200/0] via 10.1.1.1, 00:03:27
```

2. Why is this ping still not replied to?

When SanJose1 pings ISP1B, SanJose1 uses its closest IP interface, which, in this case, is 192.168.1.5. Note that the 192.168.1.4 /30 network is not yet in ISP1B's table, so ISP1B cannot reply to 192.168.1.5. There are several different ways to correct this. One approach is to have ISP1A send a summary address to ISP1B.

For the purposes of this lab, assume that AS 100 includes subnets from both the 192.168.0.0 and 192.168.1.0 address space. You need to configure BGP to propagate a supernet route that will advertise this fact. Configure SanJose1, as shown here:

```
SanJose1(config)#router bgp 100
SanJose1(config-router)#aggregate-address 192.168.0.0 255.255.254.0
```

After you complete this configuration, check SanJose1's routing table:

```

      172.16.0.0/24 is subnetted, 1 subnets
B       172.16.1.0 [20/0] via 192.168.1.6, 00:07:05
      192.168.1.0/24 is variably subnetted, 2 subnets, 2 masks
C       192.168.1.32/27 is directly connected, FastEthernet0/0
C       192.168.1.4/30 is directly connected, Serial0/1
B       192.168.0.0/23 [200/0] via 0.0.0.0, 00:00:39, Null0
```

3. From where is the route to 192.168.0.0 /23 sourced?

4. What interface is this route mapped to? Explain.

At ISP1A, issue the **clear ip bgp *** command to reset the BGP tables. Wait several seconds, and then check ISP1B's routing table.

5. Does ISP1B receive the summary route?

Verify that the supernet configuration is working. From SanJose1, ping ISP1B's FastEthernet interface, 172.16.1.2. This ping should, at last, be successful.