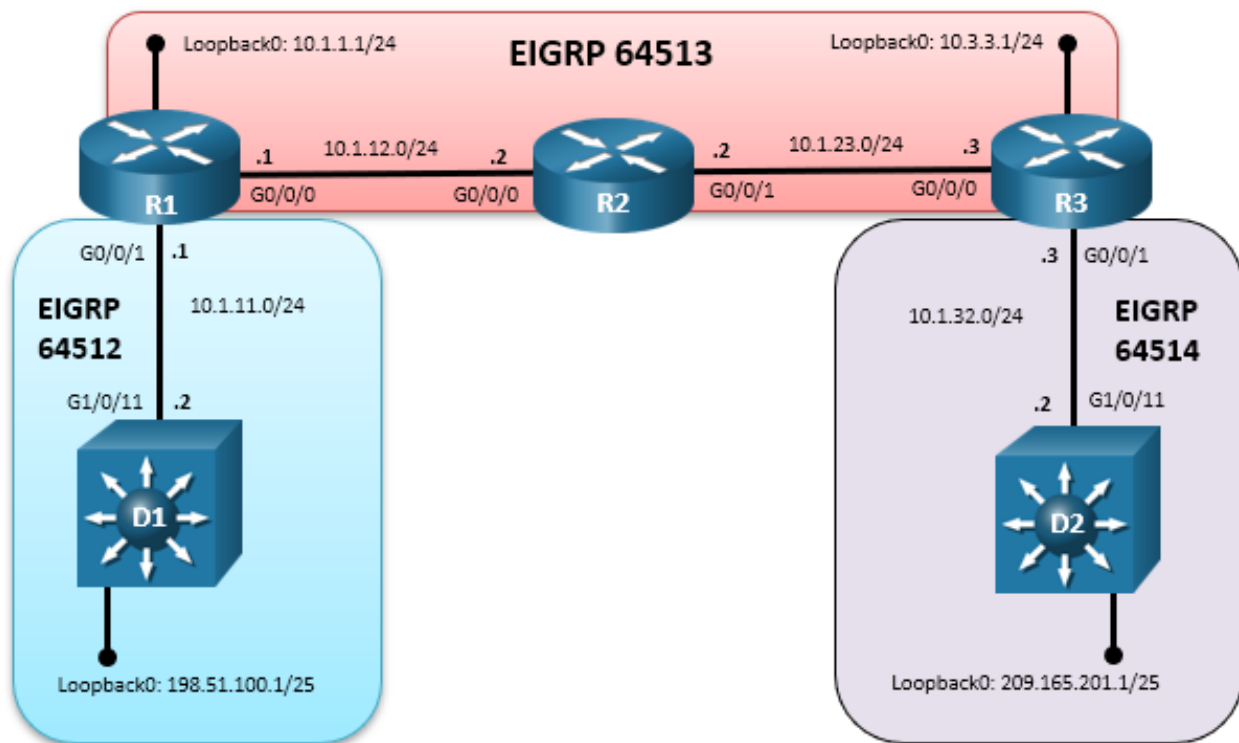# Lab - Configure Route Redistribution Within the Same Interior Gateway Protocol

Topology

Addressing Table

Device	Interface	IP Address	Subnet Mask
R1	G0/0/0	10.1.12.1	255.255.255.0
	G0/0/1	10.1.11.1	255.255.255.0
	Loopback 0	10.1.1.1	255.255.255.0
R2	G0/0/0	10.1.12.2	255.255.255.0
	G0/0/1	10.1.23.2	255.255.255.0
R3	G0/0/0	10.1.23.3	255.255.255.0
	G0/0/1	10.1.32.1	255.255.255.0
	Loopback 0	10.3.3.3	255.255.255.0
D1	G1/0/11	10.1.11.2	255.255.255.0
	Loopback 0	198.51.100.1	255.255.255.128

Device	Interface	IP Address	Subnet Mask
D2	G1/0/11	10.1.32.2	255.255.255.0
	Loopback 0	209.165.201.1	255.255.255.128

Objectives

Part 1: Build the Network and Configure Basic Device Settings

Part 2: Configure and Verify Two-Way Redistribution on R1

Part 3: Configure and Verify Two-Way Redistribution on R3

Part 4: Filter and Verify Redistribution using a Distribute List and Prefix List

Background / Scenario

Redistribution always includes two routing protocols: a source protocol and a destination protocol. The source protocol provides the network prefixes that are to be redistributed. The destination protocol receives the source protocol network prefixes. The redistribution configuration exists under the destination protocol. Examples of source protocols are static, connected, RIP, EIGRP, OSPF, IS-IS, and BGP.

Routes can be redistributed between different routing protocols or between different processes of the same routing protocol.

In this lab, you will configure mutual or two-way redistribution between multiple EIGRP processes. R1 is running classic mode EIGRP for AS 64512 and EIGRP named mode for AS 64513. R3 is running EIGRP named mode for AS 64513 and classic mode EIGRP for AS 64514.

Note: This lab is an exercise in configuring and verifying two-way route redistribution on routers R1 and R3. Route redistribution in this lab does not reflect networking best practices.

Note: The routers used with CCNP hands-on labs are Cisco 4221s with Cisco IOS XE Release 16.9.4 (universalk9 image). The switches used in the labs are Cisco Catalyst 3650s with Cisco IOS XE Release 16.9.4 (universalk9 image). Other routers, switches, and Cisco IOS versions can be used. Depending on the model and Cisco IOS version, the commands available and the output produced might vary from what is shown in the labs.

Note: Make sure that all the devices have been erased and have no startup configurations. If you are unsure, contact your instructor.

Required Resources

- 3 Routers (Cisco 4221 with Cisco IOS XE Release 16.9.4 universal image or comparable)
- 2 Switches (Cisco 3650 with Cisco IOS XE release 16.9.4 universal image or comparable)
- 1 PC (Choice of operating system with terminal emulation program installed)
- Console cables to configure the Cisco IOS devices via the console ports
- Ethernet cables as shown in the topology

Instructions

Part 1: Build the Network and Configure Basic Device Settings

In Part 1, you will set up the network topology and configure basic settings.

Step 1: Cable the network as shown in the topology.

Attach the devices as shown in the topology diagram, and cable as necessary.

Step 2: Configure basic settings for each device.

a. Console into each device, enter global configuration mode, and apply the basic settings for the lab. Initial configurations for each device are listed below.

Router R1

```
hostname R1
no ip domain lookup
banner motd # R1, Configure Route Redistribution Within the Same Interior
Gateway Protocol #
line con 0
 exec-timeout 0 0
 logging synchronous
 exit
interface g0/0/0
 ip address 10.1.12.1 255.255.255.0
 no shutdown
 exit
interface g0/0/1
 ip address 10.1.11.1 255.255.255.0
 no shutdown
 exit
interface loopback 0
 ip address 10.1.1.1 255.255.255.0
 no shutdown
 exit
router eigrp 64512
  eigrp router-id 1.1.1.1
  network 10.1.11.0 0.0.0.255
  exit
router eigrp CISCO
 address-family ipv4 unicast autonomous-system 64513
  eigrp router-id 1.1.1.1
  network 10.1.1.0 0.0.0.255
  network 10.1.12.0 0.0.0.255
  exit
 end
```

Router R2

```
hostname R2
no ip domain lookup
banner motd # R2, Configure Route Redistribution Within the Same Interior
Gateway Protocol #
line con 0
```

```
  exec-timeout 0 0
  logging synchronous
  exit
interface g0/0/0
  ip address 10.1.12.2 255.255.255.0
  no shutdown
  exit
interface g0/0/1
  ip address 10.1.23.2 255.255.255.0
  no shutdown
  exit
router eigrp CISCO
  address-family ipv4 unicast autonomous-system 64513
  eigrp router-id 2.2.2.2
   network 10.1.12.0 0.0.0.255
   network 10.1.23.0 0.0.0.255
   end
```

Router R3

```
hostname R3
no ip domain lookup
banner motd # R3, Configure Route Redistribution Within the Same Interior
Gateway Protocol #
line con 0
  exec-timeout 0 0
  logging synchronous
  exit
interface g0/0/0
  ip address 10.1.23.1 255.255.255.0
  no shutdown
  exit
interface g0/0/1
  ip address 10.1.32.1 255.255.255.0
  no shutdown
  exit
interface loopback 0
  ip address 10.3.3.1 255.255.255.0
  no shutdown
  exit
router eigrp 64514
   eigrp router-id 3.3.3.3
   network 10.1.32.0 0.0.0.255
   exit
router eigrp CISCO
  address-family ipv4 unicast autonomous-system 64513
   eigrp router-id 3.3.3.3
```

```
   network 10.1.23.0 0.0.0.255
   network 10.3.3.0 0.0.0.255
   exit
 end
```

Switch D1

```
hostname D1
no ip domain lookup
ip routing
banner motd # D1, Configure Route Redistribution Within the Same Interior
Gateway Protocol #
line con 0
 exec-timeout 0 0
 logging synchronous
 exit
interface range g1/0/1-24
 shutdown
 exit
interface g1/0/11
 no switchport
 ip address 10.1.11.2 255.255.255.0
 no shutdown
 exit
interface loopback 0
 ip address 198.51.100.1 255.255.255.128
 no shutdown
 exit
router eigrp 64512
 eigrp router-id 11.11.11.11
 network 10.1.11.0 0.0.0.255
 network 198.51.100.0 0.0.0.127
 end
```

Switch D2

```
hostname D2
no ip domain lookup
ip routing
banner motd # D2, Configure Route Redistribution Within the Same Interior
Gateway Protocol #
line con 0
 exec-timeout 0 0
 logging synchronous
 exit
interface range g1/0/1-24
 shutdown
 exit
interface g1/0/11
```

```
    no switchport
    ip address 10.1.32.2 255.255.255.0
    no shutdown
    exit
interface loopback 0
    ip address 209.165.201.1 255.255.255.128
    no shutdown
    exit
router eigrp 64514
    eigrp router-id 22.22.22.22
    network 10.1.32.0 0.0.0.255
    redistribute static
    eigrp stub static
exit
ip route 0.0.0.0 0.0.0.0 Loopback0
end
```

b. Set the clock on all devices to UTC time.

c. Save the running configuration to startup-config on all devices.

Step 3: Verify EIGRP neighborships on R1.

a. Verify that R1 has two EIGRP neighbor relationships. One EIGRP neighbor is from named mode AS 64513. The other neighbor is from EIGRP classic mode AS 64512.

```
R1# show ip eigrp neighbors
EIGRP-IPv4 Neighbors for AS(64512)
H   Address                 Interface          Hold Uptime    SRTT   RTO  QSeq
                                               (sec)          (ms)        CntNum
0   10.1.11.2               Gi0/0/1            10 00:02:10    3     100   02
EIGRP-IPv4 VR(CISCO) Address-Family Neighbors for AS(64513)
H   Address                 Interface          Hold Uptime    SRTT   RTO  QSeq
                                               (sec)          (ms)        CntNum
0   10.1.12.2               Gi0/0/0            11 00:03:38    2     100   07
```

b. Next, verify that R1 has learned internal routes form each EIGRP neighbor. Issue the **show ip route eigrp** command for AS 64512, as shown. Notice the gateway of last resort is not set and the internal EIGRP route is from D1 Loopback 0.

```
R1# show ip route eigrp 64512 | begin Gateway
Gateway of last resort is not set
      198.51.100.0/25 is subnetted, 1 subnets
D        198.51.100.0
            [90/130816] via 10.1.11.2, 04:24:45, GigabitEthernet0/0/1
```

c. Issue the **show ip route eigrp** command for EIGRP named mode, as shown. Notice the gateway of last resort is not set and the internal EIGRP routes are from R1 and R3.

```
R2# show ip route eigrp 64513 | begin Gateway
Gateway of last resort is not set
      10.0.0.0/8 is variably subnetted, 6 subnets, 2 masks
D        10.1.1.0/24 [90/10880] via 10.1.12.1, 00:10:57, GigabitEthernet0/0/0
D        10.3.3.0/24 [90/10880] via 10.1.23.1, 00:00:21, GigabitEthernet0/0/1
```

d. Verify that R3 has two EIGRP neighbor relationships. First, issue the **show ip eigrp neighbors detail** command. Notice that neighbor 10.1.32.2 is an EIGRP stub neighbor advertising static routes. Notice that R3 is using EIGRP named mode for neighbor 10.1.23.2.

```
R3# show ip eigrp neighbors detail
EIGRP-IPv4 Neighbors for AS(64514)
H   Address                 Interface         Hold Uptime    SRTT   RTO   Q  Seq
                                              (sec)          (ms)         Cnt Num
0   10.1.32.2               Gi0/0/1           11 07:40:16    1      100   0  3
    Version 25.0/2.0, Retrans: 2, Retries: 0, Prefixes: 1
    < some output omitted >

    Stub Peer Advertising (STATIC ) Routes
    Suppressing queries


EIGRP-IPv4 VR(CISCO) Address-Family Neighbors for AS(64513)
H   Address                 Interface         Hold Uptime    SRTT   RTO   Q Seq
                                              (sec)          (ms)        CntNum
0   10.1.23.2               Gi0/0/0           13 07:36:50    1      100   0 39
    Version 23.0/2.0, Retrans: 1, Retries: 0, Prefixes: 2
< some output omitted >
```

e. Issue the **show ip route eigrp** command for EIGRP named mode, as shown. Notice that the two internal EIGRP routes are from AS 64513.

```
R3# show ip route eigrp 64513 | begin Gateway
Gateway of last resort is 10.1.32.2 to network 0.0.0.0


      10.0.0.0/8 is variably subnetted, 8 subnets, 2 masks
D        10.1.1.0/24 [90/16000] via 10.1.23.2, 00:34:20, GigabitEthernet0/0/0
D        10.1.12.0/24 [90/15360] via 10.1.23.2, 00:34:20, GigabitEthernet0/0/0
```

f. Issue the **show ip route eigrp** command for AS 64514, as shown. Notice that R3 has learned a default from the EIGRP stub neighbor.

```
R3# show ip route eigrp 64514 | begin Gateway
Gateway of last resort is 10.1.32.2 to network 0.0.0.0

D*EX  0.0.0.0/0 [170/130816] via 10.1.32.2, 07:47:51, GigabitEthernet0/0/1
```

g. From R3, ping the Loopback 0 interface on D2. The ping should be successful.

```
R3# ping 209.165.201.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 209.165.201.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/2 ms
```

h. From R3, ping the Loopback 0 interface on D2 and source the ping from Loopback 0, as shown. The ping should not be successful.

```
R3# ping 209.165.201.1 source 10.3.3.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 209.165.201.1, timeout is 2 seconds:
Packet sent with a source address of 10.3.3.1
.....
```

```
Success rate is 0 percent (0/5)
```

Part 2: Configure Two-Way Redistribution on R1

In this part of the lab, you will perform EIGRP-to-EIGRP redistribution on R1. Remember that every protocol provides a seed metric at the time of redistribution. By default, source protocols, such as OSPF, RIP, IS-IS redistributed into EIGRP are given an administrative distance of 170 and a seed metric of infinity. This prevents the installation of the redistributed routes into the EIGRP topology table. However, if an EIGRP AS redistributes into another EIGRP AS, all the path metrics are preserved and included during redistribution. Therefore, setting an EIGRP seed metric in not required with performing EIGRP–to–EIGRP redistribution.

When performing redistribution, the **router** command defines the destination protocol and the **redistribute** command identifies the source protocol. For example:

```
Router(config)# router eigrp 5 !<--destination protocol
Router(config-router)# redistribute eigrp 10 !<--source protocol
```

In our example, the destination protocol is EIGRP AS 5 and the source protocol is EIGRP 10. This results in all EIGRP routes from AS 10 being redistributed into EIGRP AS 5.

Step 1: Redistribute EIGRP 64513 into EIGRP 64512.

In this step, you are going to the destination EIGRP AS 64512 to perform redistribution. The source EIGRP AS is 64513.

```
R1(config)# router eigrp 64512
R1(config-router)# redistribute eigrp 64513
R1(config-router)# exit
```

Step 2: On D1 Verify One-Way redistribution.

Issue the **show ip route eigrp** command on D1 to see the external EIGRP routes from AS 64513. Notice that the external EIGRP routes all originated from AS 64513. Also, notice that a Gateway of last resort has not been set.

```
D1# show ip route eigrp | begin Gateway
Gateway of last resort is not set

      10.0.0.0/8 is variably subnetted, 6 subnets, 2 masks
D EX     10.1.1.0/24
           [170/130816] via 10.1.11.1, 00:07:40, GigabitEthernet1/0/11
D EX     10.1.12.0/24
           [170/3072] via 10.1.11.1, 00:07:40, GigabitEthernet1/0/11
D EX     10.1.23.0/24
           [170/3328] via 10.1.11.1, 00:07:40, GigabitEthernet1/0/11
D EX     10.3.3.0/24 [170/3353] via 10.1.11.1, 00:07:40, GigabitEthernet1/0/11
```

Step 3: Redistribute EIGRP 64512 into EIGRP 64513.

Next, go to the destination AS 64513 to perform redistribution. The source EIGRP AS is 64512. To redistribute using an EIGRP named instance, you need to access the topology base, as shown.

```
R1(config)# router eigrp CISCO
R1(config-router)# address-family ipv4 unicast autonomous-system 64513
R1(config-router-af)# topology base
R1(config-router-af-topology)# redistribute eigrp 64512
R1(config-router-af-topology)# end
```

Step 4: On R3 Verify Two-Way Redistribution on R1.

Issue the **show ip route eigrp 64513 | section D EX** on D3 to see the external EIGRP routes from AS 64512.

```
R3# show ip route eigrp 64513 | section D EX
D EX     10.1.11.0/24
             [170/20480] via 10.1.23.2, 00:12:47, GigabitEthernet0/0/0
D EX     198.51.100.0
             [170/2580480] via 10.1.23.2, 00:12:47, GigabitEthernet0/0/0
```

Part 3: Configure Two-Way Redistribution on R3

In this part of the lab, you will perform EIGRP-to-EIGRP redistribution on R3. Remember a seed metric is not required.

Step 1: Redistribute EIGRP 64513 into EIGRP 64514.

The **redistribute** command is always performed on the destination protocol. Start by accessing the EIGRP process 64514. Then redistribute the source protocol, EIGRP 64513, into the destination protocol, as shown.

```
R3(config)# router eigrp 64514
R3(config-router)# redistribute eigrp 64513
R3(config-router)# end
```

Step 2: Verify redistribution on D2.

Issue the **show ip route eigrp** command on D2 to see the external EIGRP routes from AS 64513. Notice that the highlighted external EIGRP routes originated from AS 64512 and the other four external EIGRP prefixes originating from AS 64513.

```
D2# show ip route eigrp | begin Gateway
Gateway of last resort is 0.0.0.0 to network 0.0.0.0

      10.0.0.0/8 is variably subnetted, 7 subnets, 2 masks
D EX     10.1.1.0/24 [170/3353] via 10.1.32.1, 01:34:57, GigabitEthernet1/0/11
D EX     10.1.11.0/24
             [170/3584] via 10.1.32.1, 01:34:57, GigabitEthernet1/0/11
D EX     10.1.12.0/24
             [170/3328] via 10.1.32.1, 01:34:57, GigabitEthernet1/0/11
D EX     10.1.23.0/24
             [170/3072] via 10.1.32.1, 01:34:57, GigabitEthernet1/0/11
D EX     10.3.3.0/24
             [170/130816] via 10.1.32.1, 01:34:57, GigabitEthernet1/0/11
      198.51.100.0/25 is subnetted, 1 subnets
D EX     198.51.100.0
             [170/131584] via 10.1.32.1, 01:34:57, GigabitEthernet1/0/11
```

Step 3: Redistribute EIGRP 64514 into EIGRP 64513 on R3.

Next, go to the destination AS 64513 to perform redistribution. To redistribute using an EIGRP named mode you need to access the topology base. Then, the source AS 64514 is specified using the **redistribute** command, as shown.

```
R3(config)# router eigrp CISCO
```

```
R3(config-router)# address-family ipv4 unicast autonomous-system 64513
R3(config-router-af)# topology base
R3(config-router-af-topology)# redistribute eigrp 64514
R3(config-router-af-topology)# end
```

Step 4: On D1 Verify Two-Way Redistribution on R3 and end-to-end connectivity.

a. Issue the **show ip route eigrp** command on D1 to see the external EIGRP routes. Notice that both of the highlighted external EIGRP routes originated from AS 64514.

```
D1# show ip route eigrp | begin Gateway
Gateway of last resort is 10.1.11.1 to network 0.0.0.0

D*EX  0.0.0.0/0 [170/131584] via 10.1.11.1, 00:01:28, GigabitEthernet1/0/11
      10.0.0.0/8 is variably subnetted, 7 subnets, 2 masks
D EX    10.1.1.0/24
           [170/130816] via 10.1.11.1, 04:48:32, GigabitEthernet1/0/11
D EX    10.1.12.0/24
           [170/3072] via 10.1.11.1, 04:48:32, GigabitEthernet1/0/11
D EX    10.1.23.0/24
           [170/3328] via 10.1.11.1, 04:48:32, GigabitEthernet1/0/11
D EX    10.1.32.0/24
           [170/3584] via 10.1.11.1, 00:01:28, GigabitEthernet1/0/11
D EX    10.3.3.0/24 [170/3353] via 10.1.11.1, 04:48:32, GigabitEthernet1/0/11
```

b. Next, from D1 ping the 209.165.201.1 address on D2 using the Loopback 0 address on D1. The ping should be successful. This verifies full end-to-end connectivity and successful two-way redistribution on R1 and R3.

```
D1# ping 209.165.201.1 source 198.51.100.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 209.165.201.1, timeout is 2 seconds:
Packet sent with a source address of 198.51.100.1
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 2/2/4 ms
```

Part 4: Filter and Verify Redistribution using a Distribute List and Prefix List

In this part of the lab, we will filter specific EIGRP prefixes being redistributed into AS 64512 on R1 and advertised to D1. Note that the **redistribute** command cannot directly reference a prefix list, but a route map can refer to a prefix list using the **match** command. In our example, we will bind the prefix list using a distribute list.

Step 1: Create a prefix list named FILTER and specify the action for each statement.

Only allow the default route, as well as the Loopback 0 address on R1 and R3, to be sent to D1 using the prefix list name FILTER, as shown on R1. Notice the **permit** statement allows prefixes to be advertised. The last statement, sequence 20 filters all other prefixes. If not explicitly set, the deny statement is implied. This is similar to using an ACL.

```
R1(config)# ip prefix-list FILTER seq 5 permit 0.0.0.0/0
R1(config)# ip prefix-list FILTER seq 10 permit 10.1.1.0/24
R1(config)# ip prefix-list FILTER seq 15 permit 10.3.3.0/24
R1(config)# ip prefix-list FILTER seq 20 deny 0.0.0.0/0 le 32
```

Step 2: Apply the IP Prefix List using the Distribute List command in EIGRP 64512.

Next, apply the prefix-list FILTER to the distribute-list which filters routing advertisements to D1. The **out** keyword in the **distribute-list** command specifies that subnets matching prefix list FILTER will be filtered as the routing updates exit the GigabitEthernet0/0/1 interface toward D1. Using the keyword **in** would filter routes entering the routing table.

```
R1(config)# router eigrp 64512
R1(config-router)# distribute-list prefix FILTER out GigabitEthernet0/0/1
R1(config-router)# end
```

Step 3: Verify Route Filtering.

a. Issue the **show ip prefix-list detail** command on R1 to verify the hit count for each sequence in the prefix list. Notice in our example the default route has 2 hits while the deny statement has 10 hits.

```
R1# show ip prefix-list detail
Prefix-list with the last deletion/insertion: FILTER
ip prefix-list FILTER:
   count: 4, range entries: 1, sequences: 5 - 20, refcount: 4
   seq 5 permit 0.0.0.0/0 (hit count: 2, refcount: 1)
   seq 10 permit 10.1.1.0/24 (hit count: 6, refcount: 1)
   seq 15 permit 10.3.3.0/24 (hit count: 4, refcount: 2)
   seq 20 deny 0.0.0.0/0 le 32 (hit count: 10, refcount: 0)
```

b. Issue the **show ip route eigrp** command on D1 to see the external EIGRP routes. Notice the smaller routing table on D1.

```
D1# show ip route eigrp | begin Gateway
Gateway of last resort is 10.1.11.1 to network 0.0.0.0

D*EX  0.0.0.0/0 [170/131584] via 10.1.11.1, 01:22:55, GigabitEthernet1/0/11
      10.0.0.0/8 is variably subnetted, 4 subnets, 2 masks
D EX     10.1.1.0/24
            [170/130816] via 10.1.11.1, 00:31:39, GigabitEthernet1/0/11
D EX     10.3.3.0/24 [170/3353] via 10.1.11.1, 00:31:27, GigabitEthernet1/0/11
```

c. From D1 ping the 209.165.201.1 address on D2 using the Loopback 0 address on D1. The ping should be successful. This verifies full end-to-end connectivity and successful redistribution on R1 and R3, as well as route filtering on R1.

```
D1# ping 209.165.201.1 source 198.51.100.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 209.165.201.1, timeout is 2 seconds:
Packet sent with a source address of 198.51.100.1
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 2/2/3 ms
```

Reflection Questions

1. Why is a seed metric not required when redistributing EIGRP into another EIGRP process?

2. What other source protocol(s), other than EIGRP does not require a seed metric defined for redistribution into EIGRP?

3. Which EIGRP prefixes were filtered on R1 and not sent to D1?

Router Interface Summary Table

Router Model	Ethernet Interface #1	Ethernet Interface #2	Serial Interface #1	Serial Interface #2
1800	Fast Ethernet 0/0 (F0/0)	Fast Ethernet 0/1 (F0/1)	Serial 0/0/0 (S0/0/0)	Serial 0/0/1 (S0/0/1)
1900	Gigabit Ethernet 0/0 (G0/0)	Gigabit Ethernet 0/1 (G0/1)	Serial 0/0/0 (S0/0/0)	Serial 0/0/1 (S0/0/1)
2801	Fast Ethernet 0/0 (F0/0)	Fast Ethernet 0/1 (F0/1)	Serial 0/1/0 (S0/1/0)	Serial 0/1/1 (S0/1/1)
2811	Fast Ethernet 0/0 (F0/0)	Fast Ethernet 0/1 (F0/1)	Serial 0/0/0 (S0/0/0)	Serial 0/0/1 (S0/0/1)
2900	Gigabit Ethernet 0/0 (G0/0)	Gigabit Ethernet 0/1 (G0/1)	Serial 0/0/0 (S0/0/0)	Serial 0/0/1 (S0/0/1)
4221	Gigabit Ethernet 0/0/0 (G0/0/0)	Gigabit Ethernet 0/0/1 (G0/0/1)	Serial 0/1/0 (S0/1/0)	Serial 0/1/1 (S0/1/1)
4300	Gigabit Ethernet 0/0/0 (G0/0/0)	Gigabit Ethernet 0/0/1 (G0/0/1)	Serial 0/1/0 (S0/1/0)	Serial 0/1/1 (S0/1/1)

Note: To find out how the router is configured, look at the interfaces to identify the type of router and how many interfaces the router has. There is no way to effectively list all the combinations of configurations for each router class. This table includes identifiers for the possible combinations of Ethernet and Serial interfaces in the device. The table does not include any other type of interface, even though a specific router may contain one. An example of this might be an ISDN BRI interface. The string in parenthesis is the legal abbreviation that can be used in Cisco IOS commands to represent the interface.