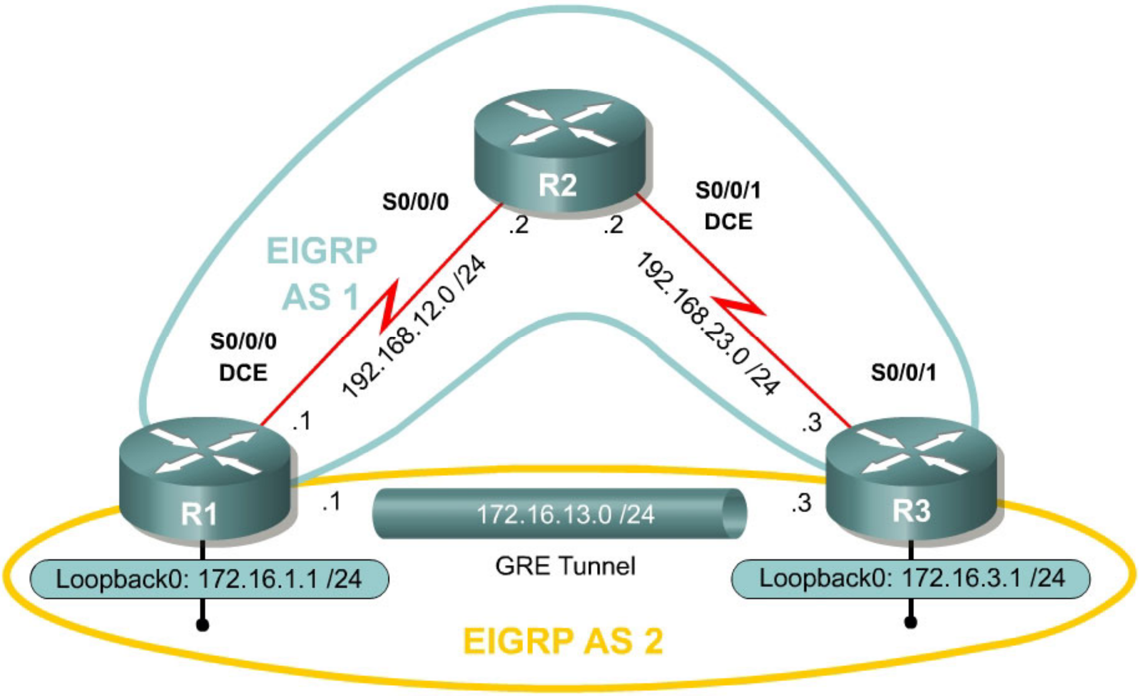
Lab 3.2 Configuring a Basic GRE Tunnel

Learning Objectives

- Configure a GRE tunnel
- Configure EIGRP on a router
- Configure and test routing over the tunnel interfaces

Topology Diagram



Scenario

This lab is designed as an introduction to tunnels. In later labs you will configure more advanced tunnels using encryption, but this lab shows the basic mechanics of GRE tunnels.

Step 1: Configure Loopbacks and Physical Interfaces

Configure the loopback interfaces with the addresses shown in the diagram. Also configure the serial interfaces shown in the diagram. Do not forget to set the clockrates on the appropriate interfaces and issue the **no shutdown** command on all serial connections. Verify that you have connectivity across the local subnet using the **ping** command. Do not set up the tunnel interface until the next step.


```
R1(config)# interface loopback 0
R1(config-if)# ip address 172.16.1.1 255.255.255.0
R1(config-if)# interface serial 0/0/0
R1(config-if)# ip address 192.168.12.1 255.255.255.0
R1(config-if)# clockrate 64000
R1(config-if)# no shutdown

R2(config)# interface serial 0/0/0
R2(config-if)# ip address 192.168.12.2 255.255.255.0
R2(config-if)# no shutdown
R2(config-if)# interface serial 0/0/1
R2(config-if)# ip address 192.168.23.2 255.255.255.0
R2(config-if)# clockrate 64000
R2(config-if)# no shutdown

R3(config)# interface loopback 0
R3(config-if)# ip address 172.16.3.1 255.255.255.0
R3(config-if)# interface serial 0/0/1
R3(config-if)# ip address 192.168.23.3 255.255.255.0
R3(config-if)# no shutdown
```

Step 2: Configure EIGRP AS 1

Configure EIGRP AS 1 for the major networks 192.168.12.0/24 and 192.168.23.0/24. Do not include the networks in the diagram falling in the 172.16.0.0/16 range. The Class C networks will serve as the transit networks for the tunnel network. Make sure you disable EIGRP automatic summarization.

```
R1(config)# router eigrp 1
R1(config-router)# no auto-summary
R1(config-router)# network 192.168.12.0

R2(config)# router eigrp 1
R2(config-router)# no auto-summary
R2(config-router)# network 192.168.12.0
R2(config-router)# network 192.168.23.0

R3(config)# router eigrp 1
R3(config-router)# no auto-summary
R3(config-router)# network 192.168.23.0
```

Verify that R1 and R3 can see the remote transit network with the **show ip route** command. If they cannot see the remote transit network, troubleshoot. R2 will not learn any new routes because it is directly connected to both networks.

```
R1# show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

C    192.168.12.0/24 is directly connected, Serial0/0/0
     172.16.0.0/24 is subnetted, 1 subnets
```



```
C       172.16.1.0 is directly connected, Loopback0
D    192.168.23.0/24 [90/2681856] via 192.168.12.2, 00:00:15, Serial0/0/0

R2# show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

C    192.168.12.0/24 is directly connected, Serial0/0/0
C    192.168.23.0/24 is directly connected, Serial0/0/1

R3# show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

D    192.168.12.0/24 [90/2681856] via 192.168.23.2, 00:00:36, Serial0/0/1
     172.16.0.0/24 is subnetted, 1 subnets
C       172.16.3.0 is directly connected, Loopback0
C    192.168.23.0/24 is directly connected, Serial0/0/1
```

Step 3: Configure a GRE Tunnel

Tunnels allow connectivity between remote areas of a network to communicate via a common network protocol and link independent of the native network protocol or routing protocol of their interconnection. For instance, consider a company with two locations in which each of the sites connects directly to the Internet with a static IP address. In order to allow private connections between the two sites, you could easily configure a tunnel between the two remote IP addresses so that private and/or encrypted communications could be sent between the two sites.

In this scenario, router R2 represents the agency providing connectivity between the two sites. R1 and R3 represent the remote sites. A tunnel will allow R1 and R3 to have a virtual private network (VPN) with each other and route between them. This type of VPN built on GRE encapsulation is not encrypted by default, but can be encrypted through simple configuration techniques.

When this configuration is complete, R2 does not need to be informed of the private networks behind R1 or R3, but simply passes IP data traffic between them based on the IP addresses on the packets it is sent. Since tunneled traffic is encapsulated within another IP header in this situation, R2 makes routing decisions based on the outermost IP header only. By running a routing protocol over the tunnel between the two sites, you can ensure that remote sites dynamically learn which remote IP networks are accessible to them.

In this lab, you use a tunnel to establish a VPN between the routers, and then route traffic between the remote sites using the tunnel interface. You will be using a base configuration without any encryption, although we will use encryption in later labs. In a production network, you would not want to send private network information through the public internet unencrypted because traffic sniffers along the way are easily able to read unencrypted data traffic.

A tunnel is a logical interface that acts as a logical connection between two endpoints. It is similar to a loopback interface in that it is a *virtual* interface created in software, but not represented by a hardware device. It is different than a loopback interface, however, in that more than one router is involved. You must configure each of the routers at the endpoints of a tunnel with a tunnel interface. GRE stands for generic routing encapsulation, and it is the simplest type of tunnel you can configure.

From global configuration mode, issue the **interface tunnel** *number* command. For simplicity, use tunnel number 0 on both routers. Next, configure an IP address with **ip address** *address subnet-mask*, just like you would do on any other interface. This IP address is used inside the tunnel, part of the private network between R1 and R3.

Finally, assign a source and destination address for the tunnel with **tunnel source** *address* and **tunnel destination** *address*, respectively. Source can also be specified by interface. These addresses specify the endpoints of the router. GRE traffic will be encapsulated out of the serial address and deencapsulated on the remote destination serial address. We do not need to configure a tunnel mode because the default tunnel mode is GRE.

```
R1(config)# int tunnel0
R1(config-if)# tunnel source serial0/0/0
R1(config-if)# tunnel destination 192.168.23.3
R1(config-if)# ip address 172.16.13.1 255.255.255.0

R3(config)# int tunnel0
R3(config-if)# tunnel source serial0/0/1
R3(config-if)# tunnel destination 192.168.12.1
R3(config-if)# ip address 172.16.13.3 255.255.255.0
```

Verify that you can **ping** across the tunnel to the other side. If you can do this, you have successfully set up the tunnel.

```
R1# ping 172.16.13.3

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.13.3, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 68/69/72 ms
```



```
R3# ping 172.16.13.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.13.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 68/68/72 ms
```

1. What is the GRE header source address of the packet?

   The source address is 192.168.12.1.

2. What is the GRE header destination address of the packet?

   The destination address is 192.168.23.3.

3. What is the source address of the packet encapsulated in the GRE tunnel?

   172.16.13.1

4. What is the destination address of the packet encapsulated in the GRE tunnel?

   172.16.13.3

5. Are these packets encrypted using the commands you entered?

   No. GRE tunnels do not have a default encryption.

Step 4: Routing EIGRP AS 2 over the Tunnel

Now that you have the tunnel set up, you can set up dynamic routing protocols over it. When the next hop address of a destination network is through the tunnel, the packet is encapsulated in an IP packet as described in the previous step.

Configure EIGRP AS 2 to route the entire 172.16.0.0 major network over the tunnel, but disable automatic summarization. Remember that R2 is not participating in this routing process so it will not need to be configured.

```
R1(config)# router eigrp 2
R1(config-router)# no auto-summary
R1(config-router)# network 172.16.0.0

R3(config)# router eigrp 2
R3(config-router)# no auto-summary
R3(config-router)# network 172.16.0.0
```

You should see EIGRP neighbors come up with their messages logged to the console. Now issue the **show ip eigrp neighbors 2** command on R1 and R3. The ‘2’ at the end of the command string specifies the AS number. If you omit this, you will get neighbor tables for both EIGRP processes.


```
R1# show ip eigrp neighbors 2
IP-EIGRP neighbors for process 2
H   Address                 Interface       Hold Uptime   SRTT   RTO  Q  Seq
                                            (sec)         (ms)       Cnt Num
0   172.16.13.3             Tu0               10 00:01:14  100   5000  0  3

R3# show ip eigrp neighbors 2
IP-EIGRP neighbors for process 2
H   Address                 Interface       Hold Uptime   SRTT   RTO  Q  Seq
                                            (sec)         (ms)       Cnt Num
0   172.16.13.1             Tu0               13 00:02:47 1608   5000  0  2
```

Notice that the neighbor adjacencies are formed over the tunnel interface, even though no physical connection between the two routers exists. If you issue the **show ip route** command on the three routers, you see that R1 and R3 see each others loopbacks. Even though R2 is in the physical path, it has no knowledge of the loopback networks.

```
R1# show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

C    192.168.12.0/24 is directly connected, Serial0/0/0
     172.16.0.0/24 is subnetted, 3 subnets
C       172.16.13.0 is directly connected, Tunnel0
C       172.16.1.0 is directly connected, Loopback0
D       172.16.3.0 [90/297372416] via 172.16.13.3, 00:04:23, Tunnel0
D    192.168.23.0/24 [90/2681856] via 192.168.12.2, 03:06:16, Serial0/0/0

R2# show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

C    192.168.12.0/24 is directly connected, Serial0/0/0
C    192.168.23.0/24 is directly connected, Serial0/0/1

R3# show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route
```



```
Gateway of last resort is not set

D    192.168.12.0/24 [90/2681856] via 192.168.23.2, 03:06:54, Serial0/0/1
     172.16.0.0/24 is subnetted, 3 subnets
C       172.16.13.0 is directly connected, Tunnel0
D       172.16.1.0 [90/297372416] via 172.16.13.1, 00:05:12, Tunnel0
C       172.16.3.0 is directly connected, Loopback0
C    192.168.23.0/24 is directly connected, Serial0/0/1
```

You will also be able to **ping** the remote loopback addresses from R1 and R3. R2 will not be able to ping either, because no route to the 172.16.0.0 network exists in its routing table.

```
R1# ping 172.16.3.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.3.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 68/68/68 ms

R2# ping 172.16.1.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.1.1, timeout is 2 seconds:
.....
Success rate is 0 percent (0/5)
R2# ping 172.16.3.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.3.1, timeout is 2 seconds:
.....
Success rate is 0 percent (0/5)

R3# ping 172.16.1.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.1.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 68/68/68 ms
```

Why can’t R2 ping 172.16.1.1 or 172.16.3.1?

These subnets are part of the private network between R1 and R3 and bridged by the tunnel interface. R2 is only aware of the public networks reachable by the ISP, which consists of only the 192.168.12.0/24 and 192.168.23.0/24 networks.

Appendix A: TCL Script Output

```
tclsh

foreach address {
172.16.1.1
172.16.3.1
172.16.13.1
172.16.13.3
192.168.12.1
```



```
192.168.12.2
192.168.23.2
192.168.23.3
} {
ping $address }

R1# tclsh
R1(tcl)#
R1(tcl)#foreach address {
+>(tcl)#172.16.1.1
+>(tcl)#172.16.3.1
+>(tcl)#172.16.13.1
+>(tcl)#172.16.13.3
+>(tcl)#192.168.12.1
+>(tcl)#192.168.12.2
+>(tcl)#192.168.23.2
+>(tcl)#192.168.23.3
+>(tcl)#} {
+>(tcl)#ping $address }

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.1.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/4 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.3.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 68/68/72 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.13.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.13.3, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 68/69/72 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.12.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 56/56/60 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.12.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 28/28/32 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.23.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 28/28/28 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.23.3, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 56/56/56 ms
R1(tcl)#tclquit

R2#tclsh
R2(tcl)#
R2(tcl)#foreach address {
+>(tcl)#172.16.1.1
+>(tcl)#172.16.3.1
+>(tcl)#172.16.13.1
+>(tcl)#172.16.13.3
+>(tcl)#192.168.12.1
+>(tcl)#192.168.12.2
```



```
+>(tcl)#192.168.23.2
+>(tcl)#192.168.23.3
+>(tcl)#} {
+>(tcl)#ping $address }

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.1.1, timeout is 2 seconds:
.....
Success rate is 0 percent (0/5)
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.3.1, timeout is 2 seconds:
.....
Success rate is 0 percent (0/5)
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.13.1, timeout is 2 seconds:
.....
Success rate is 0 percent (0/5)
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.13.3, timeout is 2 seconds:
.....
Success rate is 0 percent (0/5)
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.12.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 28/28/32 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.12.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 52/56/64 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.23.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 56/59/64 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.23.3, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 28/29/36 ms
R2(tcl)#tclquit

R3#tclsh
R3(tcl)#
R3(tcl)#foreach address {
+>(tcl)#172.16.1.1
+>(tcl)#172.16.3.1
+>(tcl)#172.16.13.1
+>(tcl)#172.16.13.3
+>(tcl)#192.168.12.1
+>(tcl)#192.168.12.2
+>(tcl)#192.168.23.2
+>(tcl)#192.168.23.3
+>(tcl)#} {
+>(tcl)#ping $address }

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.1.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 68/69/72 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.3.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.13.1, timeout is 2 seconds:
```



```
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 68/68/72 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.13.3, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/4 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.12.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 56/56/56 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.12.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 28/28/28 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.23.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 28/28/32 ms
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.23.3, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 56/59/64 ms
R3(tcl)#tclquit
```

Final Configurations

```
R1# show run
hostname R1
!
interface Tunnel0
 ip address 172.16.13.1 255.255.255.0
 tunnel source Serial0/0/0
 tunnel destination 192.168.23.3
!
interface Loopback0
 ip address 172.16.1.1 255.255.255.0
!
interface Serial0/0/0
 ip address 192.168.12.1 255.255.255.0
 clock rate 64000
 no shutdown
!
router eigrp 1
 network 192.168.12.0
 no auto-summary
!
router eigrp 2
 network 172.16.0.0
 no auto-summary
!
end

R2# show run
hostname R2
!
interface Serial0/0/0
 ip address 192.168.12.2 255.255.255.0
 no shutdown
!
interface Serial0/0/1
 ip address 192.168.23.2 255.255.255.0
 clock rate 64000
 no shutdown
```



```
!
router eigrp 1
 network 192.168.12.0
 network 192.168.23.0
 no auto-summary
!
end

R3# show run
hostname R3
!
interface Loopback0
 ip address 172.16.3.1 255.255.255.0
!
interface Tunnel0
 ip address 172.16.13.3 255.255.255.0
 tunnel source Serial0/0/1
 tunnel destination 192.168.12.1
!
interface Serial0/0/1
 ip address 192.168.23.3 255.255.255.0
 no shutdown
!
router eigrp 1
 network 192.168.23.0
 no auto-summary
!
router eigrp 2
 network 172.16.0.0
 no auto-summary
!
end
```

