Sadiq S.

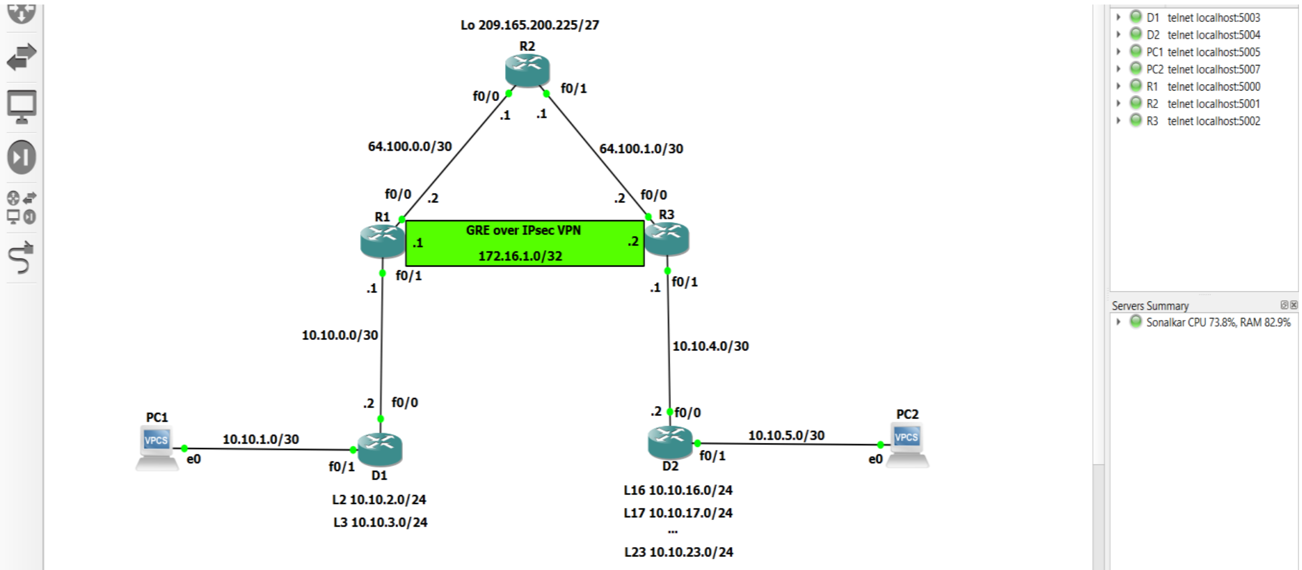
Practical 10

AIM: Implement IPsec Site-to Site VPNs
1. Implement GRE over IPsec Site -to Site VPNs

What is IP Sec VPN?

- IPsec is a group of protocols that are used together to set up encrypted connections between devices.
- Within the term "IPsec," "IP" stands for "Internet Protocol" and "sec" for "secure."
- The Internet Protocol is the main routing protocol used on the Internet; it designates where data will go using IP addresses.
- IPsec is secure because it adds encryption* and authentication to this process.
- IPsec is often used to set up VPNs, and it works by encrypting IP packets, along with authenticating the source where the packets come from.
- Users can access an IPsec VPN by logging into a VPN application, or "client." This typically requires the user to have installed the application on their device.
- VPN logins are usually password-based. While data sent over a VPN is encrypted, if user passwords are compromised, attackers can log into the VPN and steal this encrypted data.
- Using two-factor authentication (2FA) can strengthen IPsec VPN security, since stealing a password alone will no longer give an attacker access.

Step 1: Build the network:



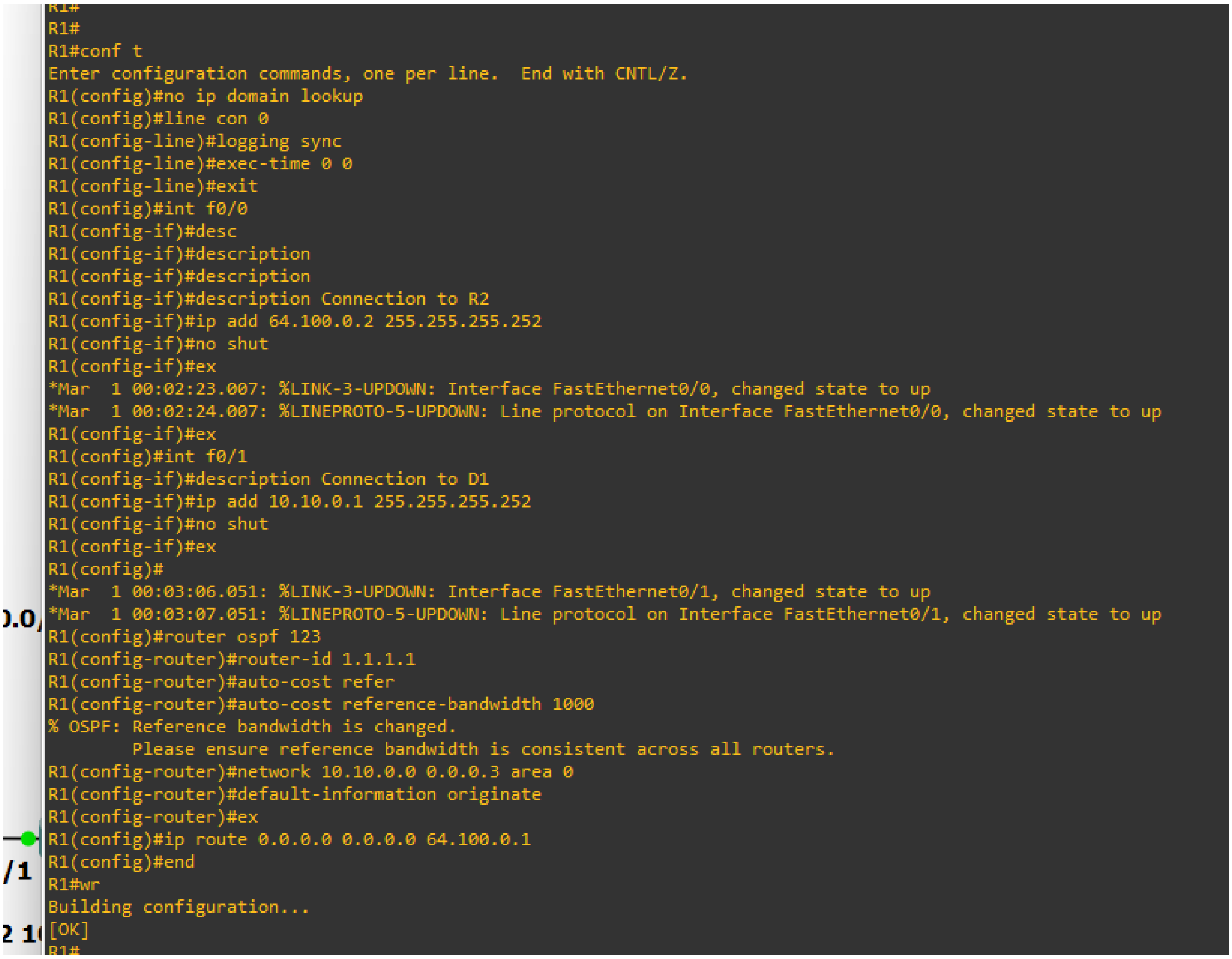
D1 telnet localhost:5003
D2 telnet localhost:5004
PC1 telnet localhost:5005
PC2 telnet localhost:5007
R1 telnet localhost:5000
R2 telnet localhost:5001
R3 telnet localhost:5002

Servers Summary

Sonalkar CPU 73.8%, RAM 82.9%

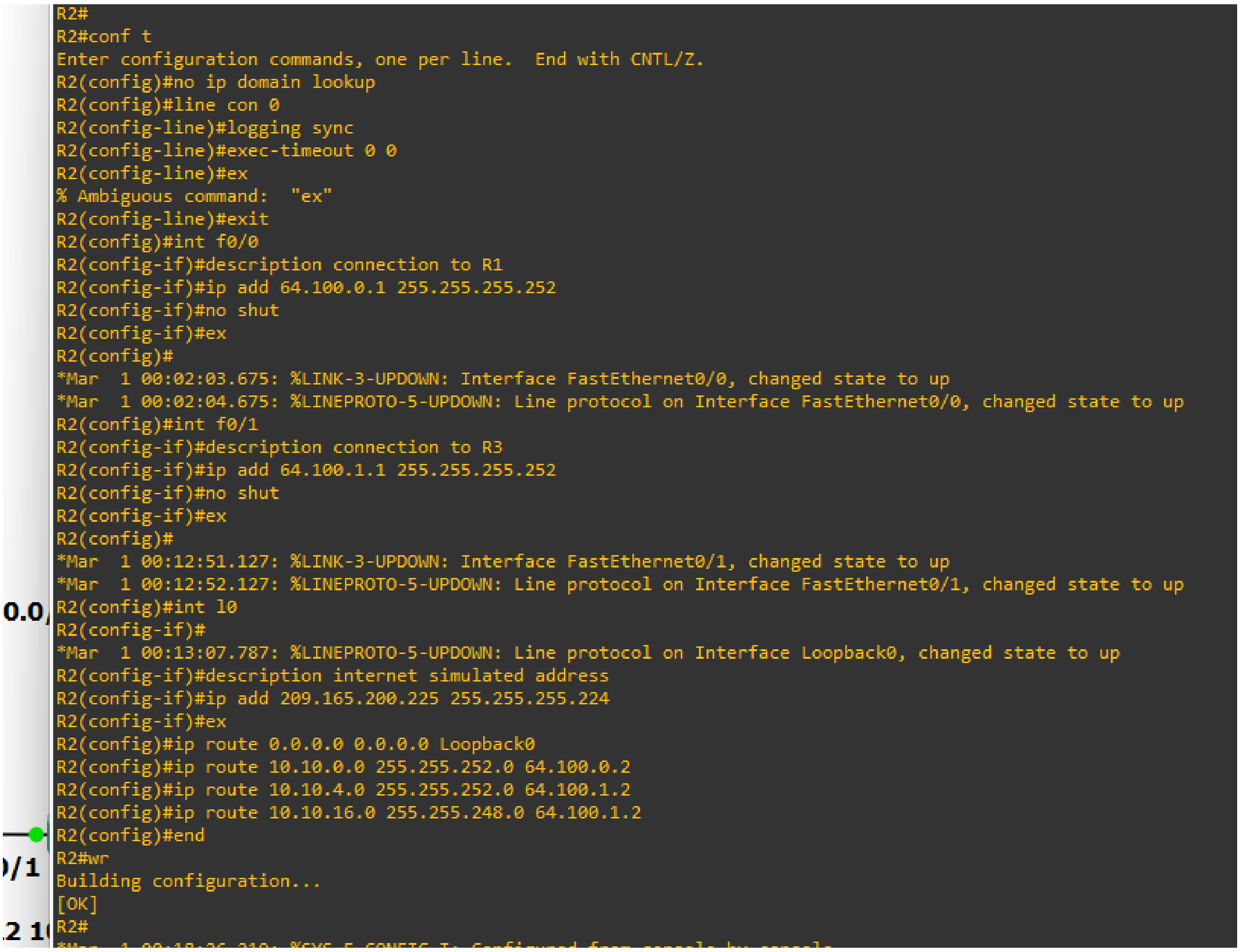
Step 2: Configure the Routers and PCs.

R1:



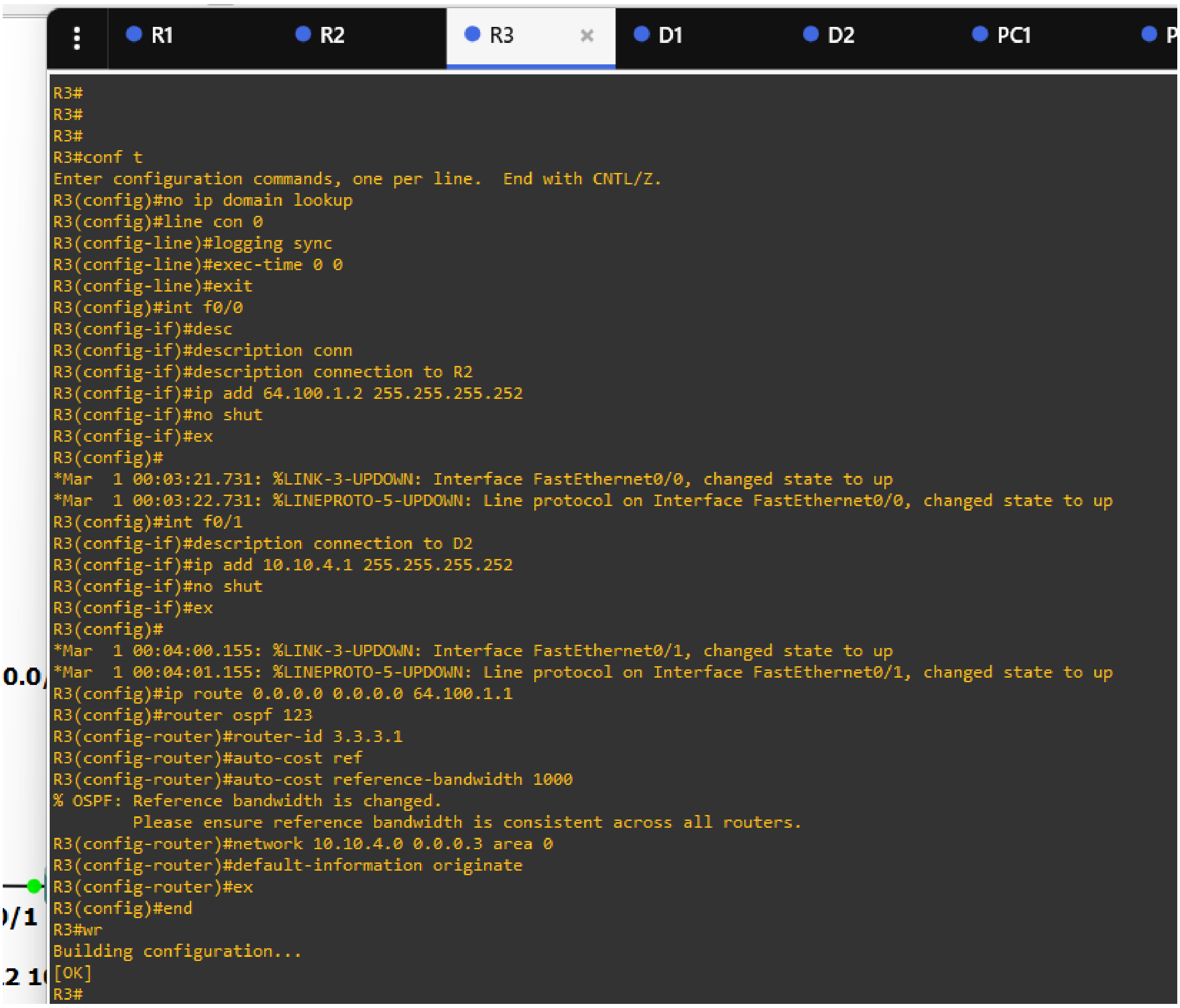
```
R1#
R1#
R1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#no ip domain lookup
R1(config)#line con 0
R1(config-line)#logging sync
R1(config-line)#exec-time 0 0
R1(config-line)#exit
R1(config)#int f0/0
R1(config-if)#desc
R1(config-if)#description
R1(config-if)#description
R1(config-if)#description Connection to R2
R1(config-if)#ip add 64.100.0.2 255.255.255.252
R1(config-if)#no shut
R1(config-if)#ex
*Mar  1 00:02:23.007: %LINK-3-UPDOWN: Interface FastEthernet0/0, changed state to up
*Mar  1 00:02:24.007: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0, changed state to up
R1(config-if)#ex
R1(config)#int f0/1
R1(config-if)#description Connection to D1
R1(config-if)#ip add 10.10.0.1 255.255.255.252
R1(config-if)#no shut
R1(config-if)#ex
R1(config)#
*Mar  1 00:03:06.051: %LINK-3-UPDOWN: Interface FastEthernet0/1, changed state to up
*Mar  1 00:03:07.051: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/1, changed state to up
R1(config)#router ospf 123
R1(config-router)#router-id 1.1.1.1
R1(config-router)#auto-cost refer
R1(config-router)#auto-cost reference-bandwidth 1000
% OSPF: Reference bandwidth is changed.
        Please ensure reference bandwidth is consistent across all routers.
R1(config-router)#network 10.10.0.0 0.0.0.3 area 0
R1(config-router)#default-information originate
R1(config-router)#ex
R1(config)#ip route 0.0.0.0 0.0.0.0 64.100.0.1
R1(config)#end
R1#wr
Building configuration...
[OK]
R1#
```

R2:



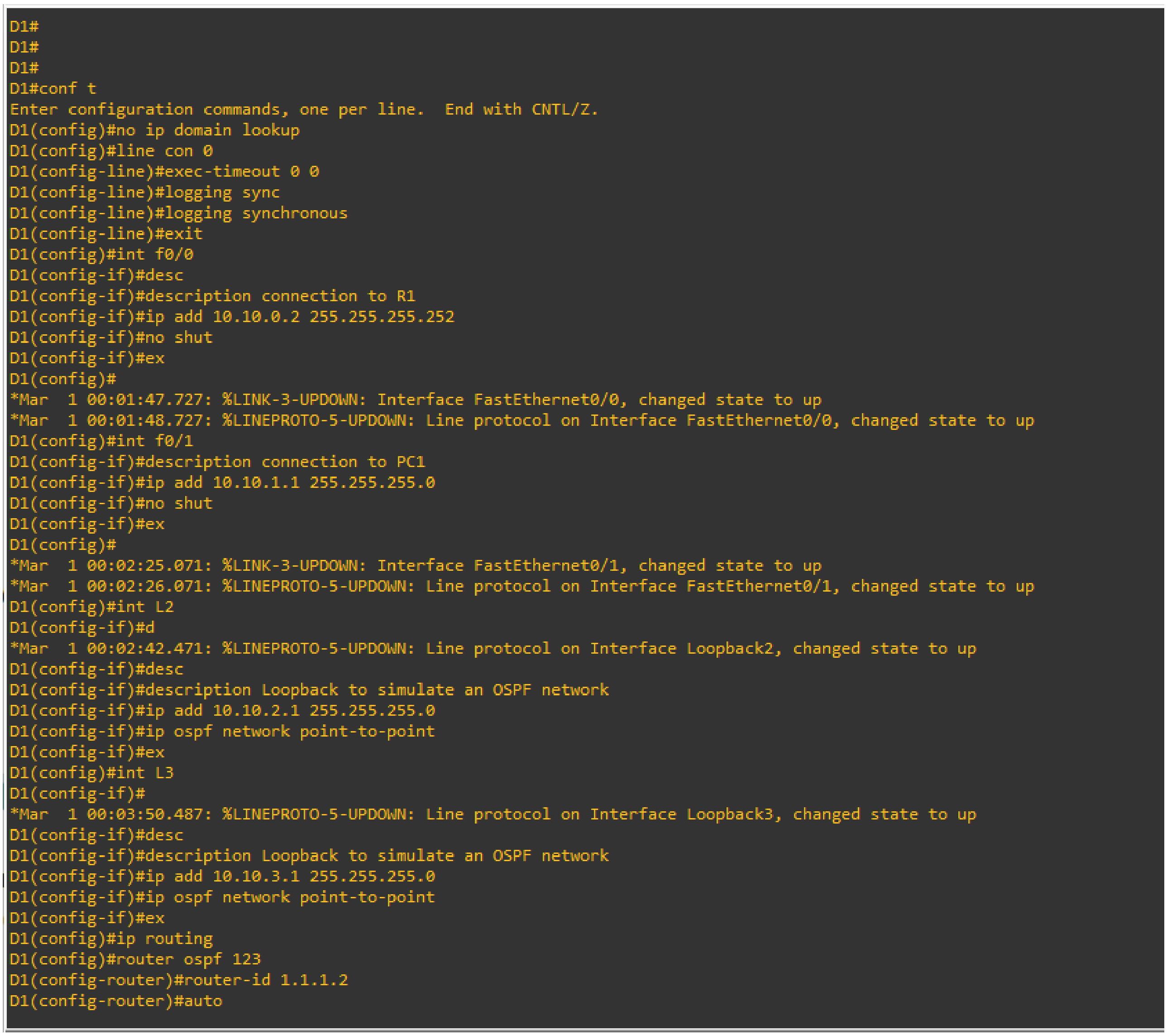
```
R2#
R2#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R2(config)#no ip domain lookup
R2(config)#line con 0
R2(config-line)#logging sync
R2(config-line)#exec-timeout 0 0
R2(config-line)#ex
% Ambiguous command:  "ex"
R2(config-line)#exit
R2(config)#int f0/0
R2(config-if)#description connection to R1
R2(config-if)#ip add 64.100.0.1 255.255.255.252
R2(config-if)#no shut
R2(config-if)#ex
R2(config)#
*Mar  1 00:02:03.675: %LINK-3-UPDOWN: Interface FastEthernet0/0, changed state to up
*Mar  1 00:02:04.675: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0, changed state to up
R2(config)#int f0/1
R2(config-if)#description connection to R3
R2(config-if)#ip add 64.100.1.1 255.255.255.252
R2(config-if)#no shut
R2(config-if)#ex
R2(config)#
*Mar  1 00:12:51.127: %LINK-3-UPDOWN: Interface FastEthernet0/1, changed state to up
*Mar  1 00:12:52.127: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/1, changed state to up
R2(config)#int l0
R2(config-if)#
*Mar  1 00:13:07.787: %LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback0, changed state to up
R2(config-if)#description internet simulated address
R2(config-if)#ip add 209.165.200.225 255.255.255.224
R2(config-if)#ex
R2(config)#ip route 0.0.0.0 0.0.0.0 Loopback0
R2(config)#ip route 10.10.0.0 255.255.252.0 64.100.0.2
R2(config)#ip route 10.10.4.0 255.255.252.0 64.100.1.2
R2(config)#ip route 10.10.16.0 255.255.248.0 64.100.1.2
R2(config)#end
R2#wr
Building configuration...
[OK]
R2#
*Mar  1 00:18:26.219: %SYS-5-CONFIG_I: Configured from console by console
```

R3:

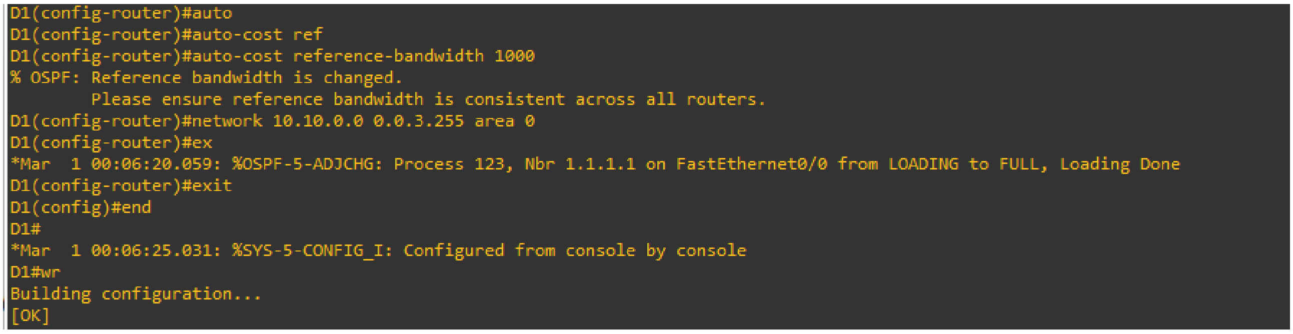


```
R3#
R3#
R3#
R3#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R3(config)#no ip domain lookup
R3(config)#line con 0
R3(config-line)#logging sync
R3(config-line)#exec-time 0 0
R3(config-line)#exit
R3(config)#int f0/0
R3(config-if)#desc
R3(config-if)#description conn
R3(config-if)#description connection to R2
R3(config-if)#ip add 64.100.1.2 255.255.255.252
R3(config-if)#no shut
R3(config-if)#ex
R3(config)#
*Mar  1 00:03:21.731: %LINK-3-UPDOWN: Interface FastEthernet0/0, changed state to up
*Mar  1 00:03:22.731: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0, changed state to up
R3(config)#int f0/1
R3(config-if)#description connection to D2
R3(config-if)#ip add 10.10.4.1 255.255.255.252
R3(config-if)#no shut
R3(config-if)#ex
R3(config)#
*Mar  1 00:04:00.155: %LINK-3-UPDOWN: Interface FastEthernet0/1, changed state to up
*Mar  1 00:04:01.155: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/1, changed state to up
R3(config)#ip route 0.0.0.0 0.0.0.0 64.100.1.1
R3(config)#router ospf 123
R3(config-router)#router-id 3.3.3.1
R3(config-router)#auto-cost ref
R3(config-router)#auto-cost reference-bandwidth 1000
% OSPF: Reference bandwidth is changed.
        Please ensure reference bandwidth is consistent across all routers.
R3(config-router)#network 10.10.4.0 0.0.0.3 area 0
R3(config-router)#default-information originate
R3(config-router)#ex
R3(config)#end
R3#wr
Building configuration...
[OK]
R3#
```

D1:

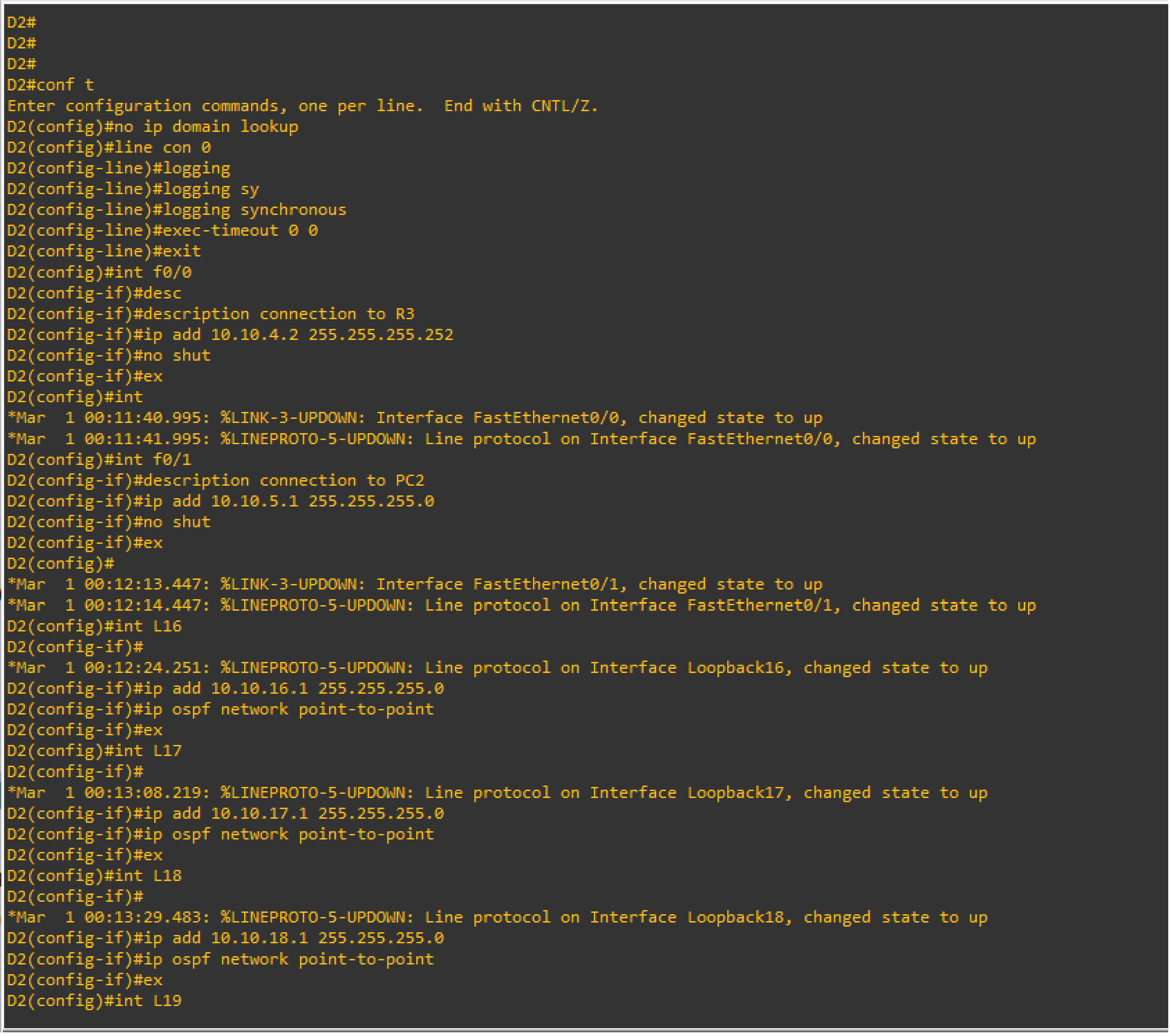


```
D1#
D1#
D1#
D1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
D1(config)#no ip domain lookup
D1(config)#line con 0
D1(config-line)#exec-timeout 0 0
D1(config-line)#logging sync
D1(config-line)#logging synchronous
D1(config-line)#exit
D1(config)#int f0/0
D1(config-if)#desc
D1(config-if)#description connection to R1
D1(config-if)#ip add 10.10.0.2 255.255.255.252
D1(config-if)#no shut
D1(config-if)#ex
D1(config)#
*Mar  1 00:01:47.727: %LINK-3-UPDOWN: Interface FastEthernet0/0, changed state to up
*Mar  1 00:01:48.727: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0, changed state to up
D1(config)#int f0/1
D1(config-if)#description connection to PC1
D1(config-if)#ip add 10.10.1.1 255.255.255.0
D1(config-if)#no shut
D1(config-if)#ex
D1(config)#
*Mar  1 00:02:25.071: %LINK-3-UPDOWN: Interface FastEthernet0/1, changed state to up
*Mar  1 00:02:26.071: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/1, changed state to up
D1(config)#int L2
D1(config-if)#d
*Mar  1 00:02:42.471: %LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback2, changed state to up
D1(config-if)#desc
D1(config-if)#description Loopback to simulate an OSPF network
D1(config-if)#ip add 10.10.2.1 255.255.255.0
D1(config-if)#ip ospf network point-to-point
D1(config-if)#ex
D1(config)#int L3
D1(config-if)#
*Mar  1 00:03:50.487: %LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback3, changed state to up
D1(config-if)#desc
D1(config-if)#description Loopback to simulate an OSPF network
D1(config-if)#ip add 10.10.3.1 255.255.255.0
D1(config-if)#ip ospf network point-to-point
D1(config-if)#ex
D1(config)#ip routing
D1(config)#router ospf 123
D1(config-router)#router-id 1.1.1.2
D1(config-router)#auto
```

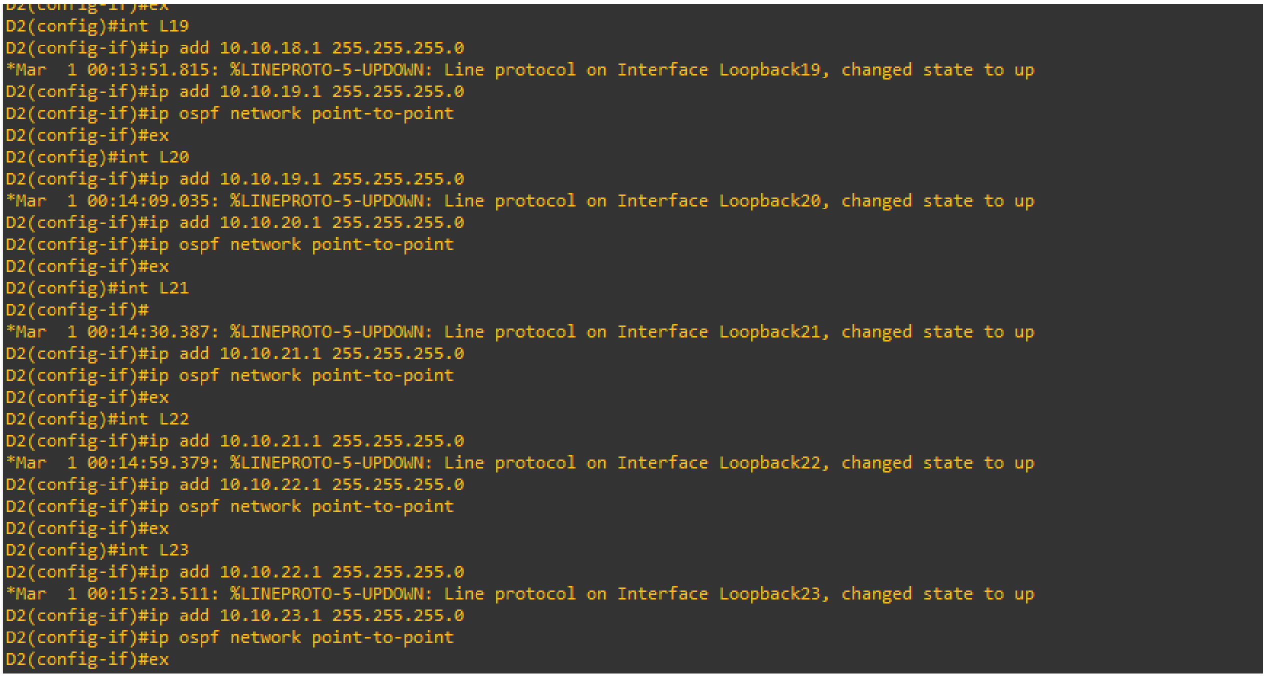


```
D1(config-router)#auto
D1(config-router)#auto-cost ref
D1(config-router)#auto-cost reference-bandwidth 1000
% OSPF: Reference bandwidth is changed.
        Please ensure reference bandwidth is consistent across all routers.
D1(config-router)#network 10.10.0.0 0.0.3.255 area 0
D1(config-router)#ex
*Mar  1 00:06:20.059: %OSPF-5-ADJCHG: Process 123, Nbr 1.1.1.1 on FastEthernet0/0 from LOADING to FULL, Loading Done
D1(config-router)#exit
D1(config)#end
D1#
*Mar  1 00:06:25.031: %SYS-5-CONFIG_I: Configured from console by console
D1#wr
Building configuration...
[OK]
```

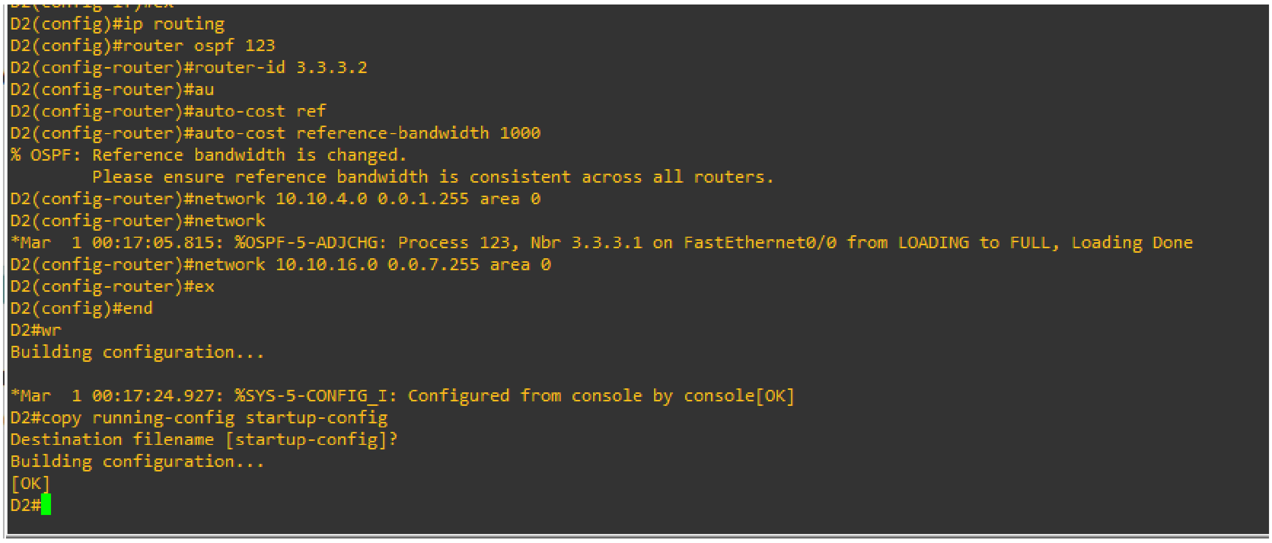
D2:



```
D2#
D2#
D2#
D2#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
D2(config)#no ip domain lookup
D2(config)#line con 0
D2(config-line)#logging
D2(config-line)#logging sy
D2(config-line)#logging synchronous
D2(config-line)#exec-timeout 0 0
D2(config-line)#exit
D2(config)#int f0/0
D2(config-if)#desc
D2(config-if)#description connection to R3
D2(config-if)#ip add 10.10.4.2 255.255.255.252
D2(config-if)#no shut
D2(config-if)#ex
D2(config)#int
*Mar  1 00:11:40.995: %LINK-3-UPDOWN: Interface FastEthernet0/0, changed state to up
*Mar  1 00:11:41.995: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0, changed state to up
D2(config)#int f0/1
D2(config-if)#description connection to PC2
D2(config-if)#ip add 10.10.5.1 255.255.255.0
D2(config-if)#no shut
D2(config-if)#ex
D2(config)#
*Mar  1 00:12:13.447: %LINK-3-UPDOWN: Interface FastEthernet0/1, changed state to up
*Mar  1 00:12:14.447: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/1, changed state to up
D2(config)#int L16
D2(config-if)#
*Mar  1 00:12:24.251: %LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback16, changed state to up
D2(config-if)#ip add 10.10.16.1 255.255.255.0
D2(config-if)#ip ospf network point-to-point
D2(config-if)#ex
D2(config)#int L17
D2(config-if)#
*Mar  1 00:13:08.219: %LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback17, changed state to up
D2(config-if)#ip add 10.10.17.1 255.255.255.0
D2(config-if)#ip ospf network point-to-point
D2(config-if)#ex
D2(config)#int L18
D2(config-if)#
*Mar  1 00:13:29.483: %LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback18, changed state to up
D2(config-if)#ip add 10.10.18.1 255.255.255.0
D2(config-if)#ip ospf network point-to-point
D2(config-if)#ex
D2(config)#int L19
```

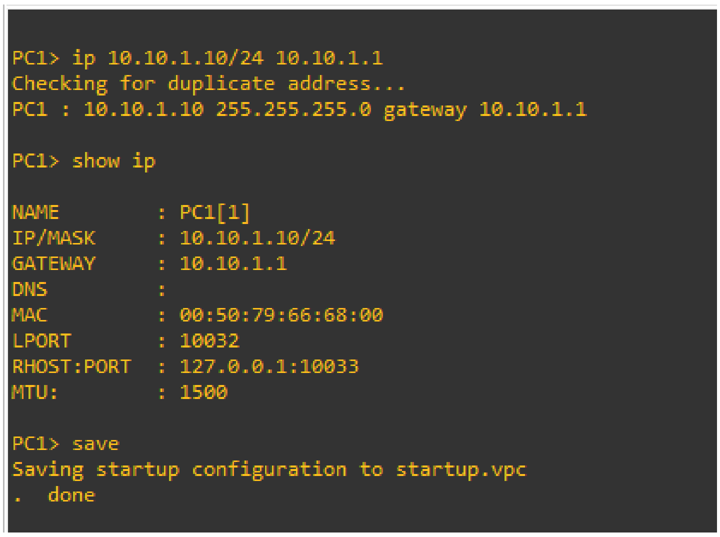


```
D2(config-if)#ex
D2(config)#int L19
D2(config-if)#ip add 10.10.18.1 255.255.255.0
*Mar  1 00:13:51.815: %LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback19, changed state to up
D2(config-if)#ip add 10.10.19.1 255.255.255.0
D2(config-if)#ip ospf network point-to-point
D2(config-if)#ex
D2(config)#int L20
D2(config-if)#ip add 10.10.19.1 255.255.255.0
*Mar  1 00:14:09.035: %LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback20, changed state to up
D2(config-if)#ip add 10.10.20.1 255.255.255.0
D2(config-if)#ip ospf network point-to-point
D2(config-if)#ex
D2(config)#int L21
D2(config-if)#
*Mar  1 00:14:30.387: %LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback21, changed state to up
D2(config-if)#ip add 10.10.21.1 255.255.255.0
D2(config-if)#ip ospf network point-to-point
D2(config-if)#ex
D2(config)#int L22
D2(config-if)#ip add 10.10.21.1 255.255.255.0
*Mar  1 00:14:59.379: %LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback22, changed state to up
D2(config-if)#ip add 10.10.22.1 255.255.255.0
D2(config-if)#ip ospf network point-to-point
D2(config-if)#ex
D2(config)#int L23
D2(config-if)#ip add 10.10.22.1 255.255.255.0
*Mar  1 00:15:23.511: %LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback23, changed state to up
D2(config-if)#ip add 10.10.23.1 255.255.255.0
D2(config-if)#ip ospf network point-to-point
D2(config-if)#ex
```



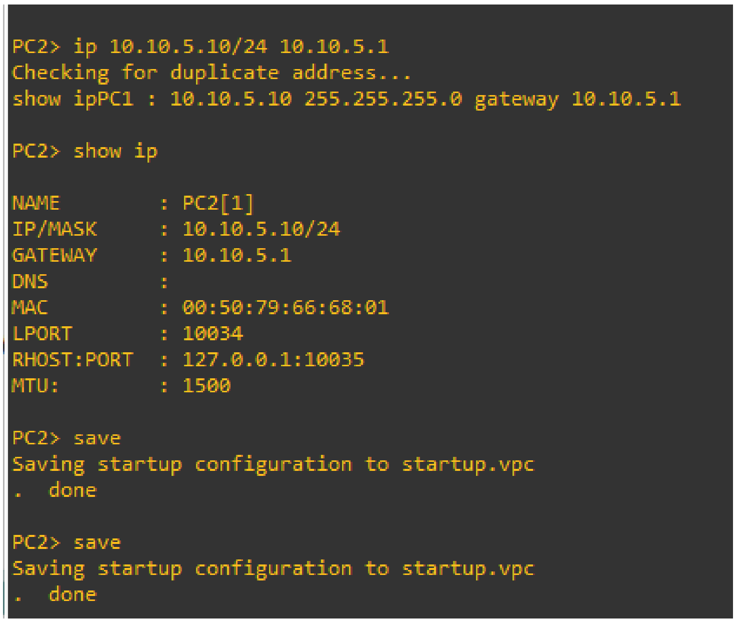
```
D2(config-if)#ex
D2(config)#ip routing
D2(config)#router ospf 123
D2(config-router)#router-id 3.3.3.2
D2(config-router)#au
D2(config-router)#auto-cost ref
D2(config-router)#auto-cost reference-bandwidth 1000
% OSPF: Reference bandwidth is changed.
        Please ensure reference bandwidth is consistent across all routers.
D2(config-router)#network 10.10.4.0 0.0.1.255 area 0
D2(config-router)#network
*Mar  1 00:17:05.815: %OSPF-5-ADJCHG: Process 123, Nbr 3.3.3.1 on FastEthernet0/0 from LOADING to FULL, Loading Done
D2(config-router)#network 10.10.16.0 0.0.7.255 area 0
D2(config-router)#ex
D2(config)#end
D2#wr
Building configuration...

*Mar  1 00:17:24.927: %SYS-5-CONFIG_I: Configured from console by console[OK]
D2#copy running-config startup-config
Destination filename [startup-config]?
Building configuration...
[OK]
D2#
```

PC1:



```
PC1> ip 10.10.1.10/24 10.10.1.1
Checking for duplicate address...
PC1 : 10.10.1.10 255.255.255.0 gateway 10.10.1.1

PC1> show ip

NAME        : PC1[1]
IP/MASK     : 10.10.1.10/24
GATEWAY     : 10.10.1.1
DNS         :
MAC         : 00:50:79:66:68:00
LPORT       : 10032
RHOST:PORT  : 127.0.0.1:10033
MTU:        : 1500

PC1> save
Saving startup configuration to startup.vpc
.  done
```

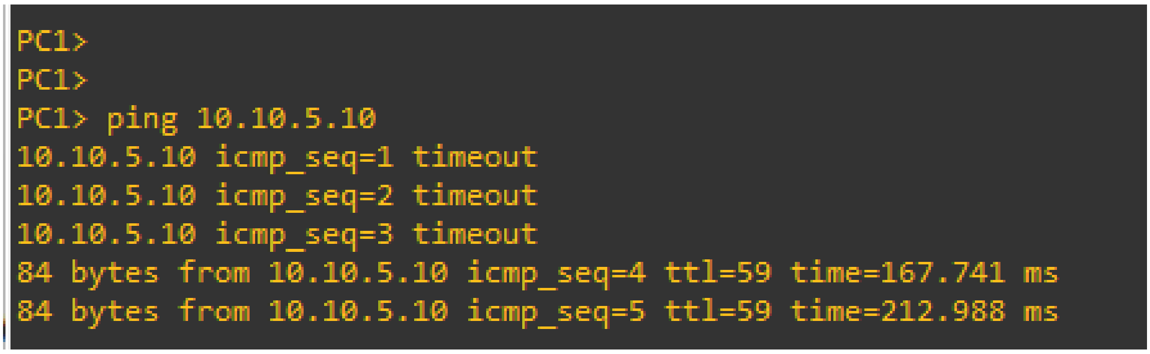
PC2:



```
PC2> ip 10.10.5.10/24 10.10.5.1
Checking for duplicate address...
show ipPC1 : 10.10.5.10 255.255.255.0 gateway 10.10.5.1

PC2> show ip

NAME        : PC2[1]
IP/MASK     : 10.10.5.10/24
GATEWAY     : 10.10.5.1
DNS         :
MAC         : 00:50:79:66:68:01
LPORT       : 10034
RHOST:PORT  : 127.0.0.1:10035
MTU:        : 1500

PC2> save
Saving startup configuration to startup.vpc
.  done

PC2> save
Saving startup configuration to startup.vpc
.  done
```

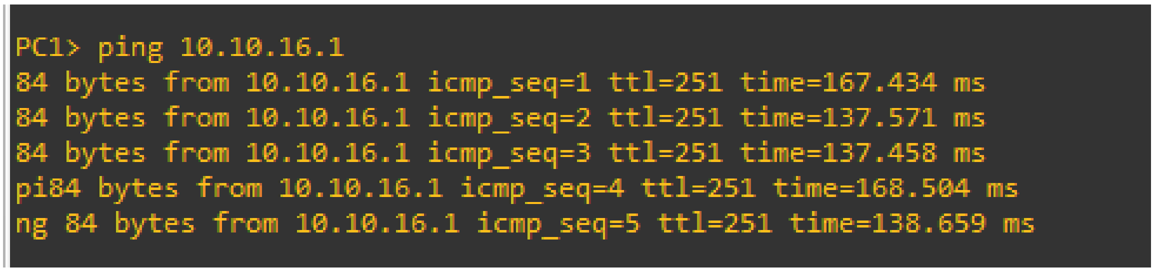
Step 3: On PC1, verify end-to-end connectivity.

From PC1, ping PC3(10.10.5.10).



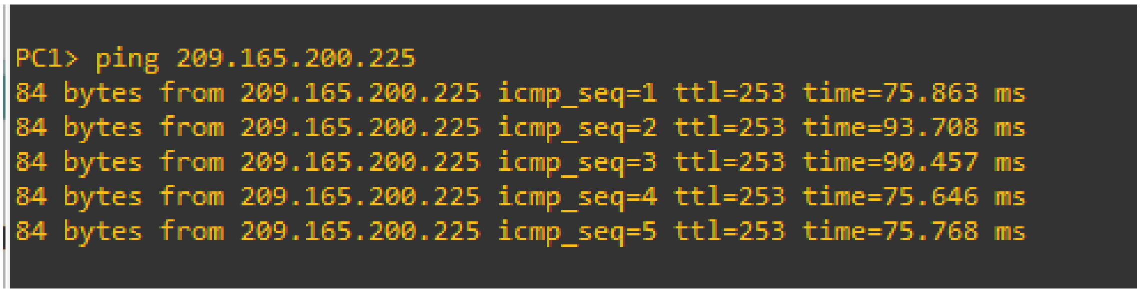
```
PC1>
PC1>
PC1> ping 10.10.5.10
10.10.5.10 icmp_seq=1 timeout
10.10.5.10 icmp_seq=2 timeout
10.10.5.10 icmp_seq=3 timeout
84 bytes from 10.10.5.10 icmp_seq=4 ttl=59 time=167.741 ms
84 bytes from 10.10.5.10 icmp_seq=5 ttl=59 time=212.988 ms
```

From PC1, ping the first loopback on D3 (10.10.16.1).



```
PC1> ping 10.10.16.1
84 bytes from 10.10.16.1 icmp_seq=1 ttl=251 time=167.434 ms
84 bytes from 10.10.16.1 icmp_seq=2 ttl=251 time=137.571 ms
84 bytes from 10.10.16.1 icmp_seq=3 ttl=251 time=137.458 ms
pi84 bytes from 10.10.16.1 icmp_seq=4 ttl=251 time=168.504 ms
ng 84 bytes from 10.10.16.1 icmp_seq=5 ttl=251 time=138.659 ms
```

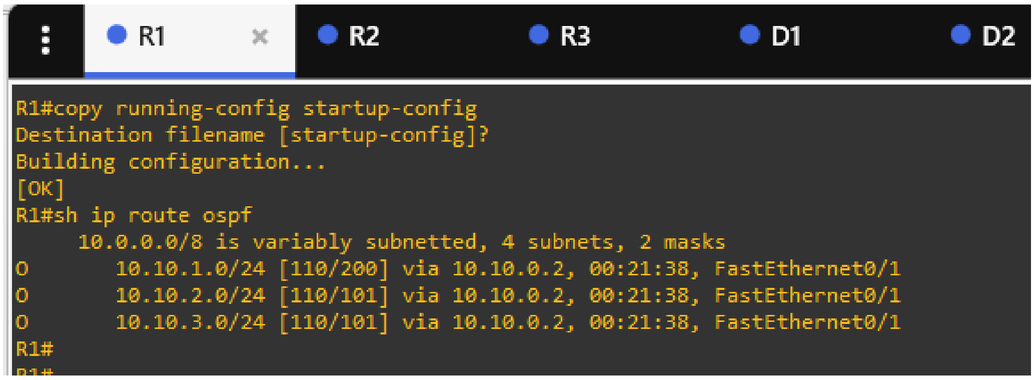
Finally, from PC1, ping the default gateway loopback on R2 (209.165.200.225).



```
PC1> ping 209.165.200.225
84 bytes from 209.165.200.225 icmp_seq=1 ttl=253 time=75.863 ms
84 bytes from 209.165.200.225 icmp_seq=2 ttl=253 time=93.708 ms
84 bytes from 209.165.200.225 icmp_seq=3 ttl=253 time=90.457 ms
84 bytes from 209.165.200.225 icmp_seq=4 ttl=253 time=75.646 ms
84 bytes from 209.165.200.225 icmp_seq=5 ttl=253 time=75.768 ms
```

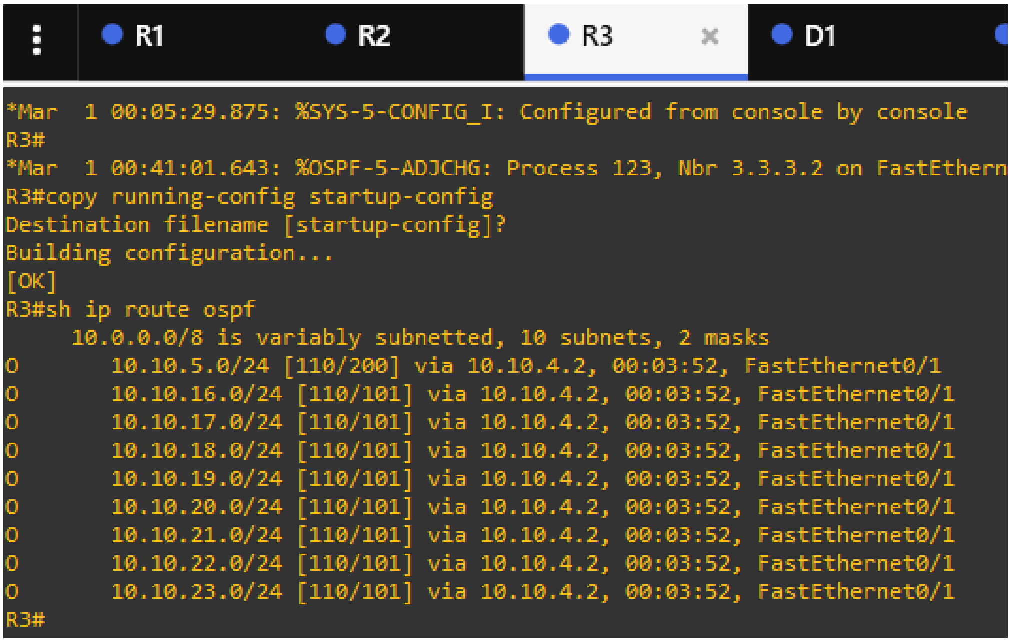
Step 4: Verify the routing table of R1 and R3.

Verify the OSPF routing table of R1.



```
R1#copy running-config startup-config
Destination filename [startup-config]?
Building configuration...
[OK]
R1#sh ip route ospf
     10.0.0.0/8 is variably subnetted, 4 subnets, 2 masks
O       10.10.1.0/24 [110/200] via 10.10.0.2, 00:21:38, FastEthernet0/1
O       10.10.2.0/24 [110/101] via 10.10.0.2, 00:21:38, FastEthernet0/1
O       10.10.3.0/24 [110/101] via 10.10.0.2, 00:21:38, FastEthernet0/1
R1#
R1#
```

Verify the OSPF routing table of R3.



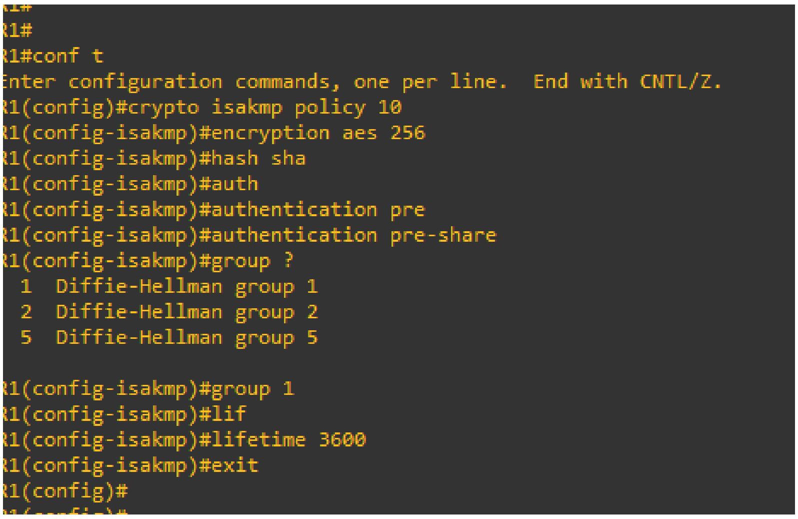
```
*Mar  1 00:05:29.875: %SYS-5-CONFIG_I: Configured from console by console
R3#
*Mar  1 00:41:01.643: %OSPF-5-ADJCHG: Process 123, Nbr 3.3.3.2 on FastEthern
R3#copy running-config startup-config
Destination filename [startup-config]?
Building configuration...
[OK]
R3#sh ip route ospf
     10.0.0.0/8 is variably subnetted, 10 subnets, 2 masks
O       10.10.5.0/24 [110/200] via 10.10.4.2, 00:03:52, FastEthernet0/1
O       10.10.16.0/24 [110/101] via 10.10.4.2, 00:03:52, FastEthernet0/1
O       10.10.17.0/24 [110/101] via 10.10.4.2, 00:03:52, FastEthernet0/1
O       10.10.18.0/24 [110/101] via 10.10.4.2, 00:03:52, FastEthernet0/1
O       10.10.19.0/24 [110/101] via 10.10.4.2, 00:03:52, FastEthernet0/1
O       10.10.20.0/24 [110/101] via 10.10.4.2, 00:03:52, FastEthernet0/1
O       10.10.21.0/24 [110/101] via 10.10.4.2, 00:03:52, FastEthernet0/1
O       10.10.22.0/24 [110/101] via 10.10.4.2, 00:03:52, FastEthernet0/1
O       10.10.23.0/24 [110/101] via 10.10.4.2, 00:03:52, FastEthernet0/1
R3#
```

Step 5: Configure GRE over IPsec using a Crypto Map on R1.
On R1, configure the ISAKMP policy and pre-shared key.

Like site-to-site VPNs using crypto maps, GRE over IPsec also requires an ISAKMP policy configuration and pre-shared key configured.
In this lab, we will use the following parameters for the ISAKMP policy 10 on R1:

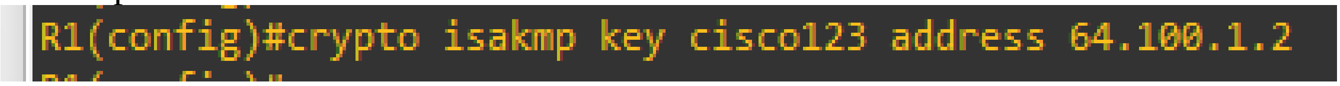
- Encryption: aes 256
- Hash: sha256
- Authentication method: pre-share key
- Diffie-Hellman group: 14
- Lifetime: 3600 seconds (60 minutes / 1 hour)

Configure ISAKMP policy 10 on R1:



```
R1#
R1#
R1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#crypto isakmp policy 10
R1(config-isakmp)#encryption aes 256
R1(config-isakmp)#hash sha
R1(config-isakmp)#auth
R1(config-isakmp)#authentication pre
R1(config-isakmp)#authentication pre-share
R1(config-isakmp)#group ?
  1  Diffie-Hellman group 1
  2  Diffie-Hellman group 2
  5  Diffie-Hellman group 5

R1(config-isakmp)#group 1
R1(config-isakmp)#lif
R1(config-isakmp)#lifetime 3600
R1(config-isakmp)#exit
R1(config)#
R1(config)#
```

Configure the pre-shared key of cisco123 on R1. This command points to the remote peer R3 G0/0/0 IP address.



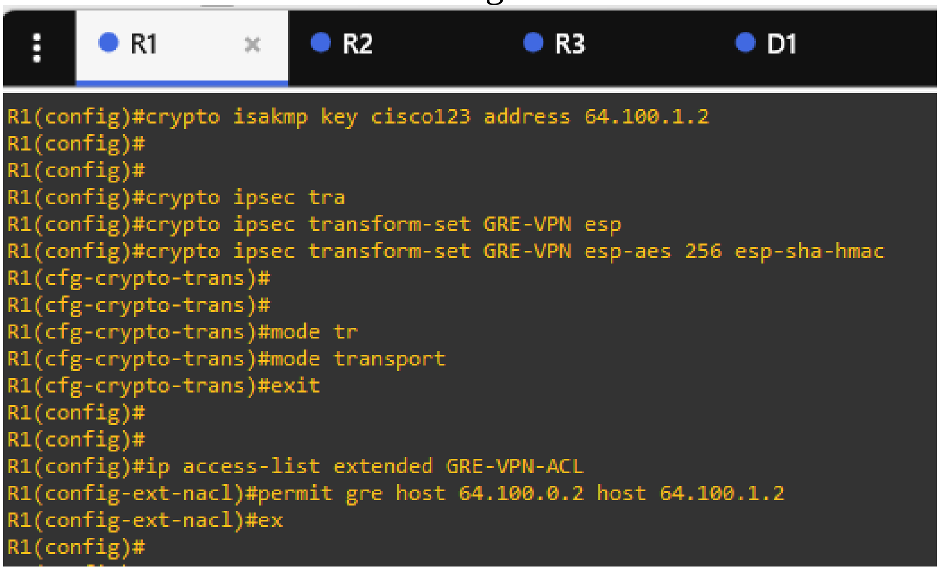
```
R1(config)#crypto isakmp key cisco123 address 64.100.1.2
```

On R1, configure the transform set and VPN ACL:

Create a transform set called GRE-VPN using AES 256 cipher with ESP and the SHA 256 hash function.

Unlike a site-to-site IPsec VPN, the transform must use transport mode. The mode command is used to identify the type of tunnel that will be established. The default is mode tunnel mode. However, GRE over IPsec should be configured using the mode transport command.

Next, create a named extended ACL called GRE-VPN-ACL that makes the tunnel interface traffic interesting.

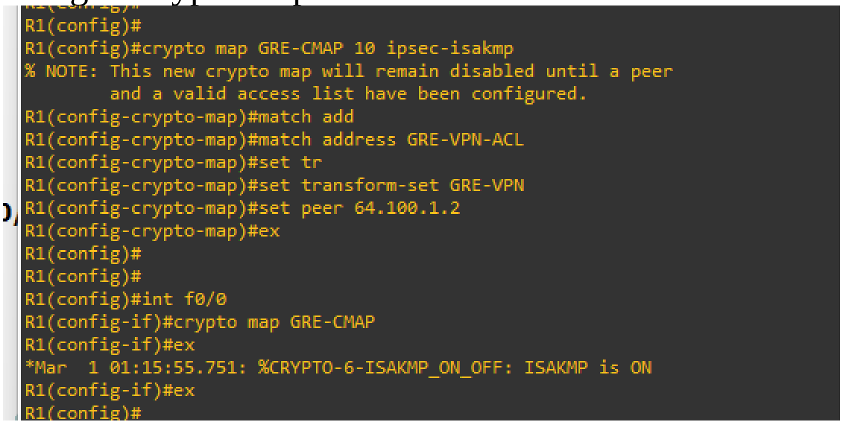


```
R1(config)#crypto isakmp key cisco123 address 64.100.1.2
R1(config)#
R1(config)#
R1(config)#crypto ipsec tra
R1(config)#crypto ipsec transform-set GRE-VPN esp
R1(config)#crypto ipsec transform-set GRE-VPN esp-aes 256 esp-sha-hmac
R1(cfg-crypto-trans)#
R1(cfg-crypto-trans)#
R1(cfg-crypto-trans)#mode tr
R1(cfg-crypto-trans)#mode transport
R1(cfg-crypto-trans)#exit
R1(config)#
R1(config)#
R1(config)#ip access-list extended GRE-VPN-ACL
R1(config-ext-nacl)#permit gre host 64.100.0.2 host 64.100.1.2
R1(config-ext-nacl)#ex
R1(config)#
```

On R1, configure the crypto map and apply it to the interface.

Create a crypto map called GRE-CMAP that associates the new GRE-VPN ACL, transform set, and peer.

Assign a crypto map called GRE-MAP on G0/0/0.

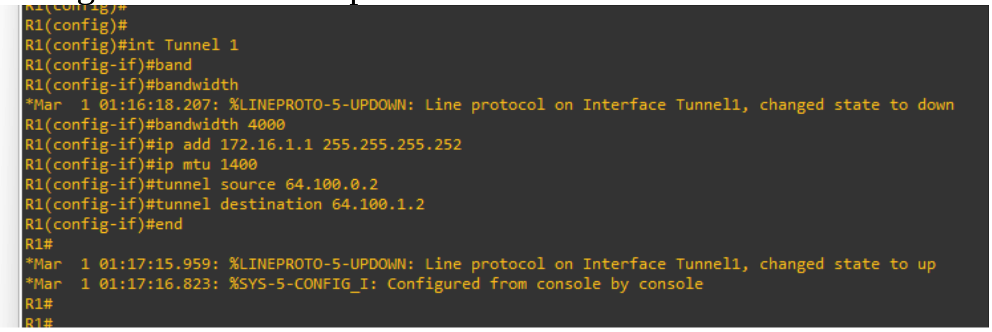


```
R1(config)#
R1(config)#crypto map GRE-CMAP 10 ipsec-isakmp
% NOTE: This new crypto map will remain disabled until a peer
        and a valid access list have been configured.
R1(config-crypto-map)#match add
R1(config-crypto-map)#match address GRE-VPN-ACL
R1(config-crypto-map)#set tr
R1(config-crypto-map)#set transform-set GRE-VPN
R1(config-crypto-map)#set peer 64.100.1.2
R1(config-crypto-map)#ex
R1(config)#
R1(config)#
R1(config)#int f0/0
R1(config-if)#crypto map GRE-CMAP
R1(config-if)#ex
*Mar  1 01:15:55.751: %CRYPTO-6-ISAKMP_ON_OFF: ISAKMP is ON
R1(config-if)#ex
R1(config)#
```

On R1, configure the GRE tunnel interface:

Configure a GRE tunnel interface as shown. To enable GRE on the tunnel interface, the tunnel mode gre ipv4 command is required.

However, this command is enabled by default and will therefore not be configured in our example.



```
R1(config)#
R1(config)#
R1(config)#int Tunnel 1
R1(config-if)#band
R1(config-if)#bandwidth
*Mar  1 01:16:18.207: %LINEPROTO-5-UPDOWN: Line protocol on Interface Tunnel1, changed state to down
R1(config-if)#bandwidth 4000
R1(config-if)#ip add 172.16.1.1 255.255.255.252
R1(config-if)#ip mtu 1400
R1(config-if)#tunnel source 64.100.0.2
R1(config-if)#tunnel destination 64.100.1.2
R1(config-if)#end
R1#
*Mar  1 01:17:15.959: %LINEPROTO-5-UPDOWN: Line protocol on Interface Tunnel1, changed state to up
*Mar  1 01:17:16.823: %SYS-5-CONFIG_I: Configured from console by console
R1#
R1#
```

Step 6: Configure GRE over IPsec using a Tunnel IPsec Profile on R3.

In this part, we will configure GRE over IPsec using tunnel IPsec profiles on R3.

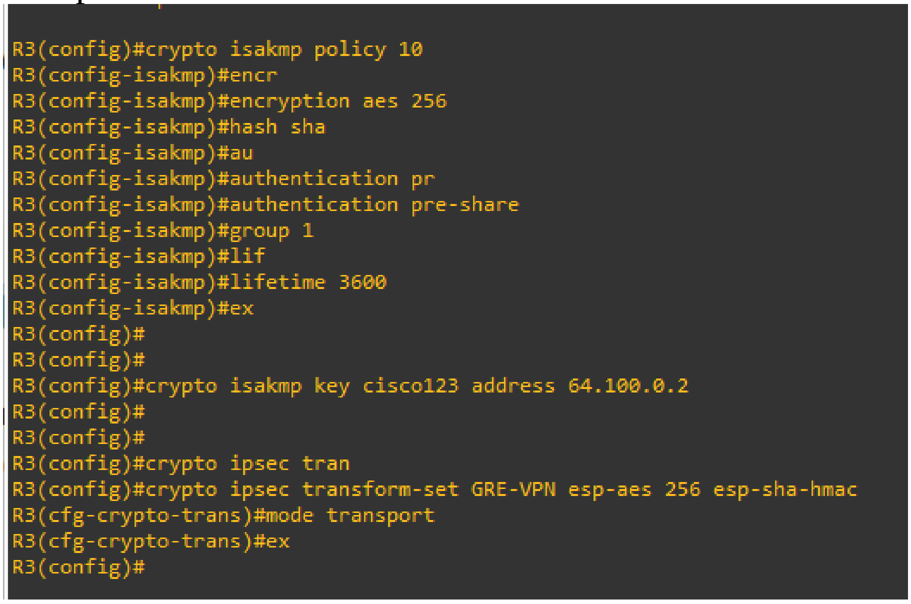
On R3, configure the ISAKMP policy, pre-shared key, and transform set.

In this step, we will configure the same parameters for the ISAKMP policy 10 that we configured on R1.

Configure ISAKMP policy 10 on R3:

Configure the pre-shared key of cisco123 on R1. This command points to the remote peer R3 G0/0/0 IP address.

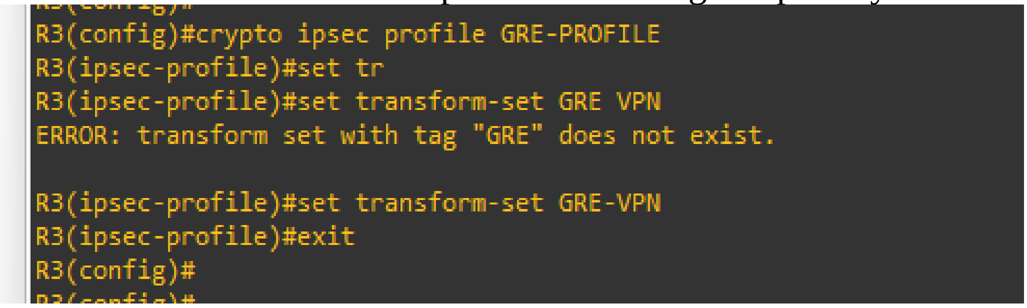
Create a new transform set called GRE-VPN using the same security parameters and transport mode that we configured on R1. Also configure the mode transport command.



```
R3(config)#crypto isakmp policy 10
R3(config-isakmp)#encr
R3(config-isakmp)#encryption aes 256
R3(config-isakmp)#hash sha
R3(config-isakmp)#au
R3(config-isakmp)#authentication pr
R3(config-isakmp)#authentication pre-share
R3(config-isakmp)#group 1
R3(config-isakmp)#lif
R3(config-isakmp)#lifetime 3600
R3(config-isakmp)#ex
R3(config)#
R3(config)#
R3(config)#crypto isakmp key cisco123 address 64.100.0.2
R3(config)#
R3(config)#
R3(config)#crypto ipsec tran
R3(config)#crypto ipsec transform-set GRE-VPN esp-aes 256 esp-sha-hmac
R3(cfg-crypto-trans)#mode transport
R3(cfg-crypto-trans)#ex
R3(config)#
```

- **On R3, configure the IPsec profile.**

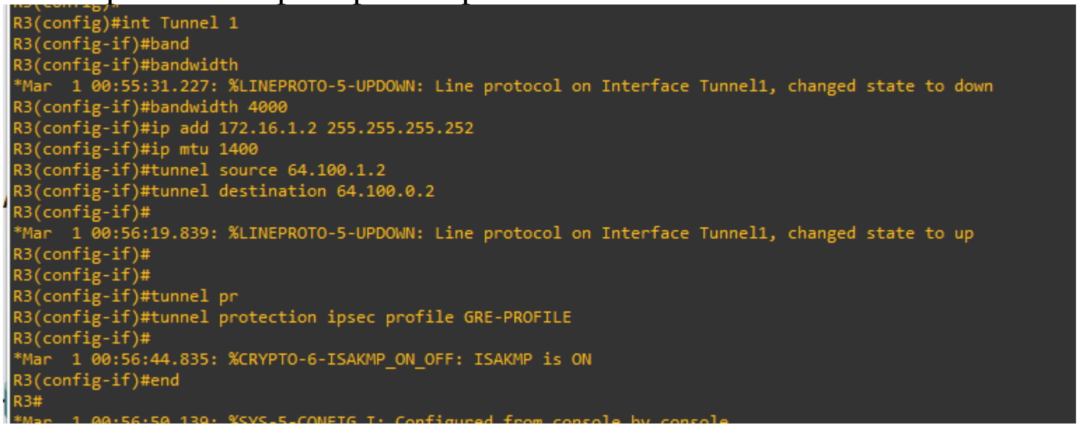
Instead of a crypto map, we will configure an IPsec profile called GRE-PROFILE using the crypto ipsec profile ipsec-profile-name global configuration command.
In IPsec profile configuration mode, specify the transform set to be negotiated using the set transform-set transform-set-name command.
Multiple transform sets can be specified in order of priority.
The fist transform-set-name specified is the highest priority.



```
R3(config)#crypto ipsec profile GRE-PROFILE
R3(ipsec-profile)#set tr
R3(ipsec-profile)#set transform-set GRE VPN
ERROR: transform set with tag "GRE" does not exist.

R3(ipsec-profile)#set transform-set GRE-VPN
R3(ipsec-profile)#exit
R3(config)#
```

- **On R3, configure the tunnel interface.**

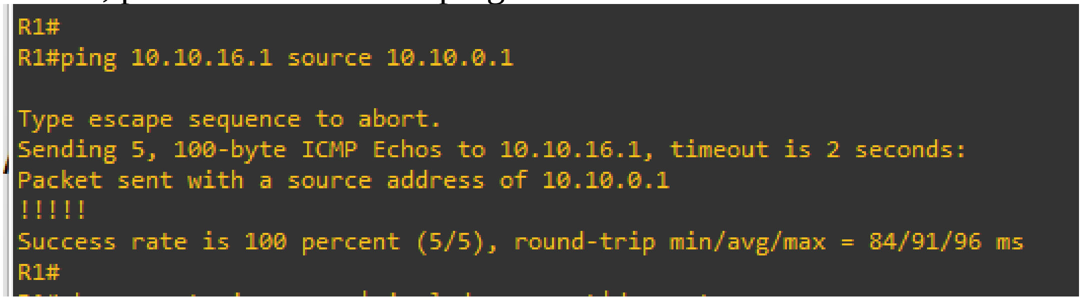
On R3, configure a GRE tunnel interface.
Apply the IPsec profile GRE-PROFILE to the Tunnel 1 interface using the tunnel protection ipsec profile profile-name command.



```
R3(config)#int Tunnel 1
R3(config-if)#band
R3(config-if)#bandwidth
*Mar  1 00:55:31.227: %LINEPROTO-5-UPDOWN: Line protocol on Interface Tunnel1, changed state to down
R3(config-if)#bandwidth 4000
R3(config-if)#ip add 172.16.1.2 255.255.255.252
R3(config-if)#ip mtu 1400
R3(config-if)#tunnel source 64.100.1.2
R3(config-if)#tunnel destination 64.100.0.2
R3(config-if)#
*Mar  1 00:56:19.839: %LINEPROTO-5-UPDOWN: Line protocol on Interface Tunnel1, changed state to up
R3(config-if)#
R3(config-if)#
R3(config-if)#tunnel pr
R3(config-if)#tunnel protection ipsec profile GRE-PROFILE
R3(config-if)#
*Mar  1 00:56:44.835: %CRYPTO-6-ISAKMP_ON_OFF: ISAKMP is ON
R3(config-if)#end
R3#
*Mar  1 00:56:50.139: %SYS-5-CONFIG_I: Configured from console by console
```

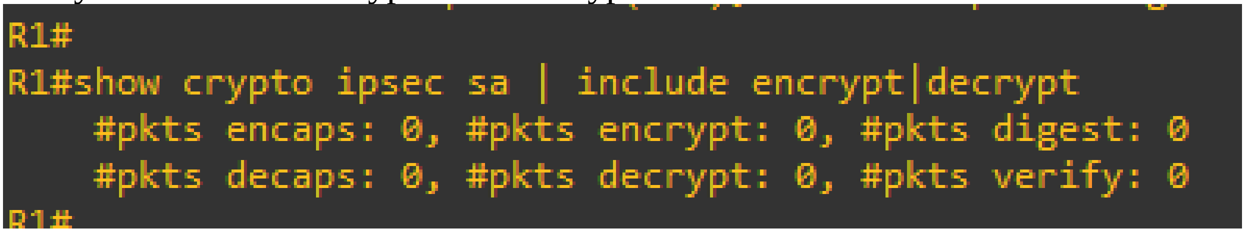
On R1 and R3, enable OSPF routing on the tunnel interface.

Verify that the GRE over IPsec VPN is operational.
On R1, perform an extended ping to the R3 10.10.16.1 interface.



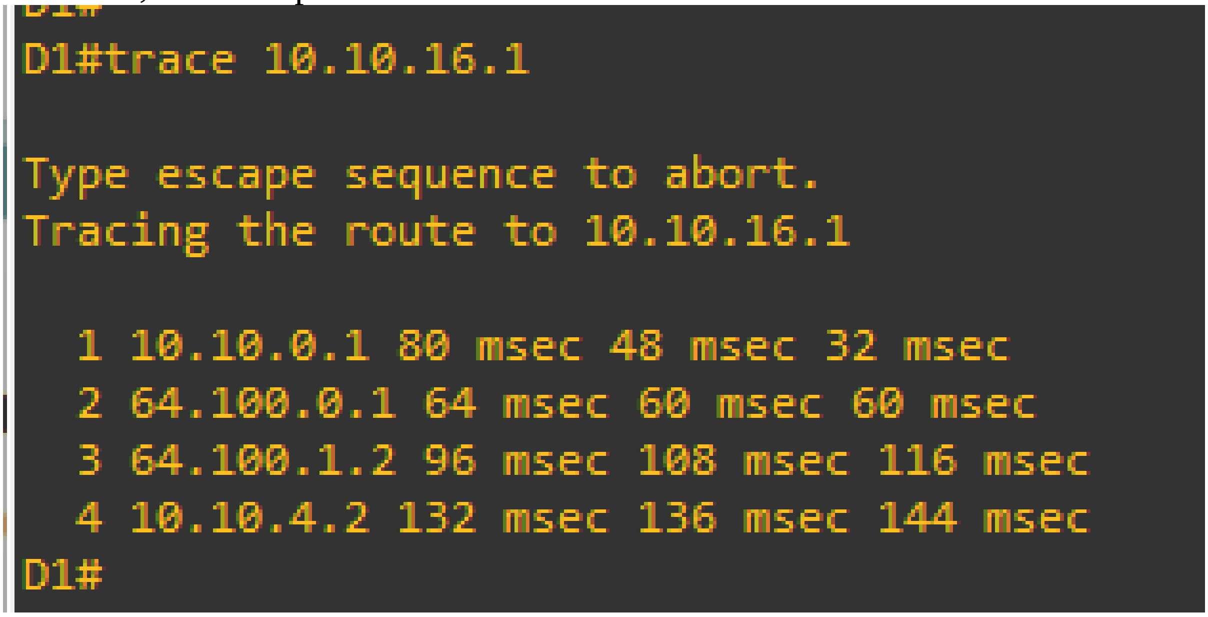
```
R1#
R1#ping 10.10.16.1 source 10.10.0.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.10.16.1, timeout is 2 seconds:
Packet sent with a source address of 10.10.0.1
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 84/91/96 ms
R1#
```

The pings are successful, and it appears that the VPN is operational. On R1, verify the IPsec SA encrypted and decrypted statistics.



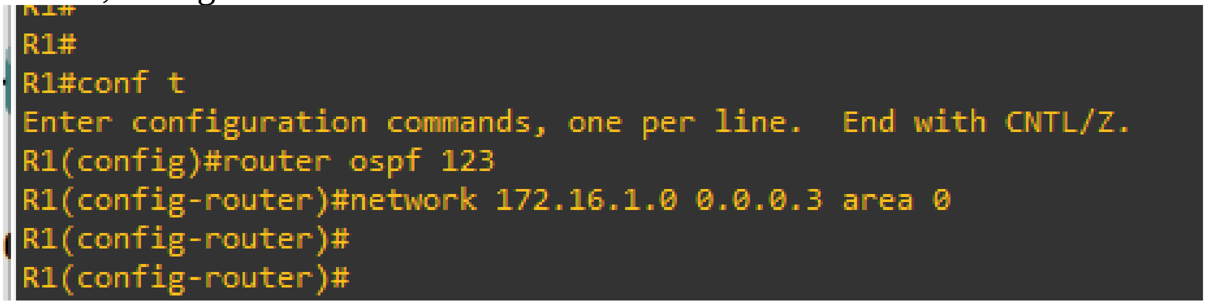
```
R1#
R1#show crypto ipsec sa | include encrypt|decrypt
    #pkts encaps: 0, #pkts encrypt: 0, #pkts digest: 0
    #pkts decaps: 0, #pkts decrypt: 0, #pkts verify: 0
R1#
```

From D1, trace the path taken to the R3 10.10.16.1 interface.



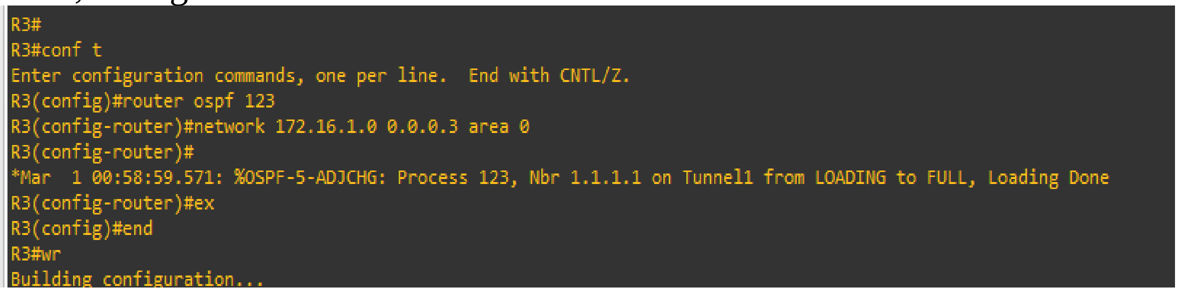
```
D1#
D1#trace 10.10.16.1

Type escape sequence to abort.
Tracing the route to 10.10.16.1

  1 10.10.0.1 80 msec 48 msec 32 msec
  2 64.100.0.1 64 msec 60 msec 60 msec
  3 64.100.1.2 96 msec 108 msec 116 msec
  4 10.10.4.2 132 msec 136 msec 144 msec
D1#
```

On R1, configure OSPF to advertise the tunnel interfaces.



```
R1#
R1#
R1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#router ospf 123
R1(config-router)#network 172.16.1.0 0.0.0.3 area 0
R1(config-router)#
R1(config-router)#
```

On R3, configure OSPF to advertise the tunnel interfaces.



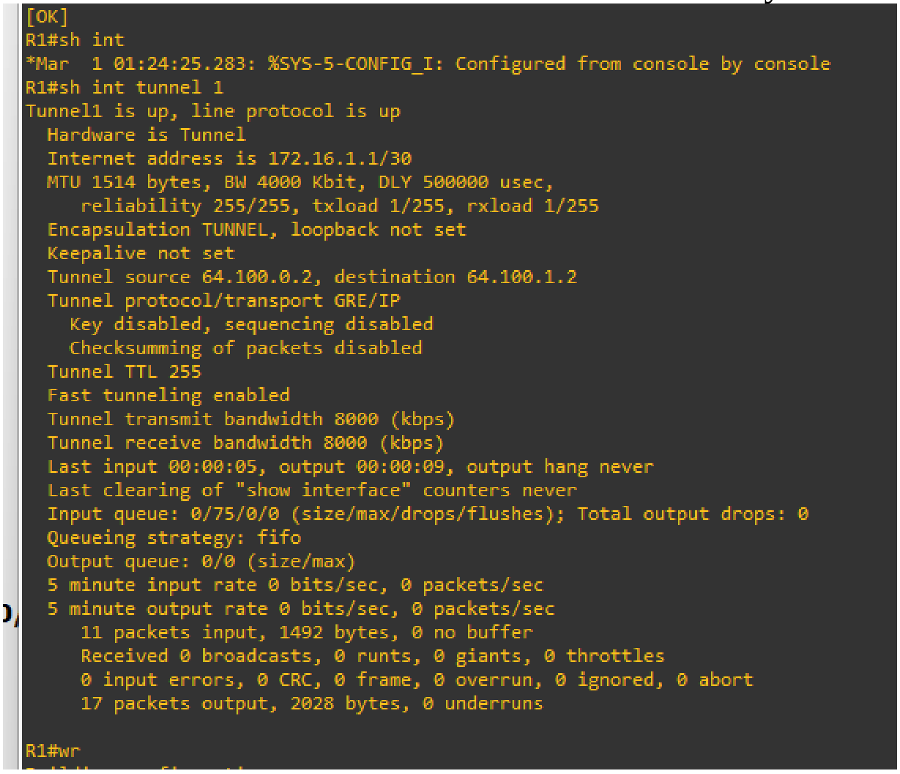
```
R3#
R3#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R3(config)#router ospf 123
R3(config-router)#network 172.16.1.0 0.0.0.3 area 0
R3(config-router)#
*Mar  1 00:58:59.571: %OSPF-5-ADJCHG: Process 123, Nbr 1.1.1.1 on Tunnel1 from LOADING to FULL, Loading Done
R3(config-router)#ex
R3(config)#end
R3#wr
Building configuration...
```

Step 7: Verify the GRE over IPsec Tunnel on R1 and R3

Now that the GRE over IPsec has been configured, we must verify that the tunnel interfaces are correctly enabled, that the crypto session is active, and then generate traffic to confirm it is traversing securely over the IPsec tunnel.

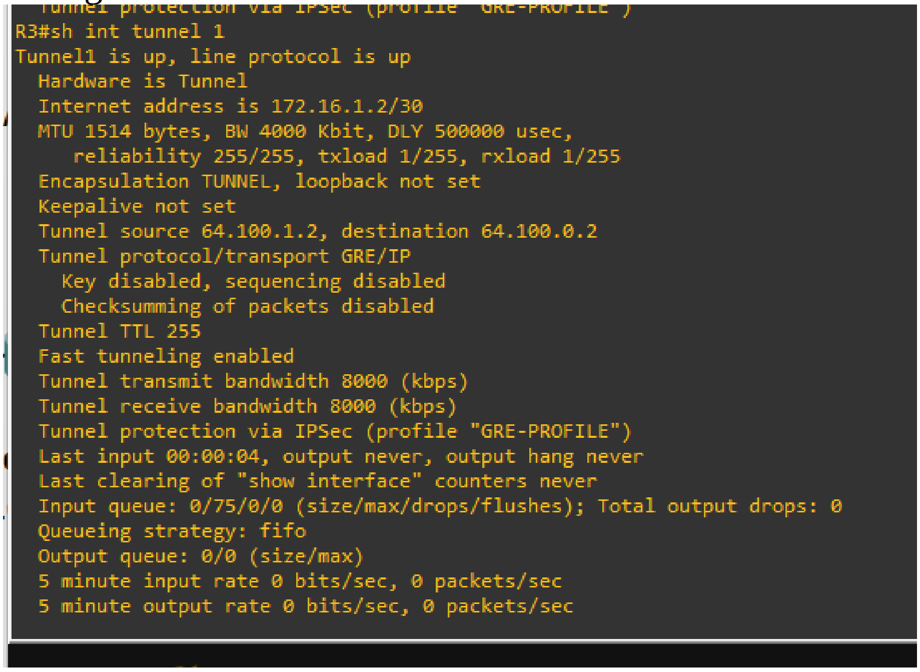
- **On R1 and R3, verify the tunnel interfaces.**

Use the show interfaces tunnel 1 command to verify the interface settings.



```
[OK]
R1#sh int
*Mar  1 01:24:25.283: %SYS-5-CONFIG_I: Configured from console by console
R1#sh int tunnel 1
Tunnel1 is up, line protocol is up
  Hardware is Tunnel
  Internet address is 172.16.1.1/30
  MTU 1514 bytes, BW 4000 Kbit, DLY 500000 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation TUNNEL, loopback not set
  Keepalive not set
  Tunnel source 64.100.0.2, destination 64.100.1.2
  Tunnel protocol/transport GRE/IP
    Key disabled, sequencing disabled
    Checksumming of packets disabled
  Tunnel TTL 255
  Fast tunneling enabled
  Tunnel transmit bandwidth 8000 (kbps)
  Tunnel receive bandwidth 8000 (kbps)
  Last input 00:00:05, output 00:00:09, output hang never
  Last clearing of "show interface" counters never
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: fifo
  Output queue: 0/0 (size/max)
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
     11 packets input, 1492 bytes, 0 no buffer
     Received 0 broadcasts, 0 runts, 0 giants, 0 throttles
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
     17 packets output, 2028 bytes, 0 underruns

R1#wr
```

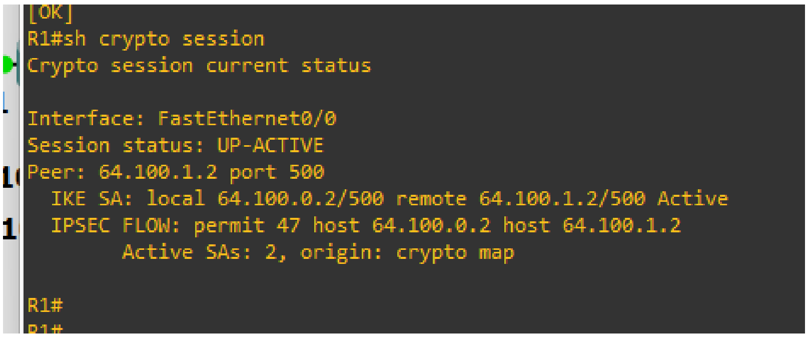
On R3, use the show interfaces tunnel 1 command to verify the interface settings.



```
  Tunnel protection via IPSec (profile "GRE-PROFILE")
R3#sh int tunnel 1
Tunnel1 is up, line protocol is up
  Hardware is Tunnel
  Internet address is 172.16.1.2/30
  MTU 1514 bytes, BW 4000 Kbit, DLY 500000 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation TUNNEL, loopback not set
  Keepalive not set
  Tunnel source 64.100.1.2, destination 64.100.0.2
  Tunnel protocol/transport GRE/IP
    Key disabled, sequencing disabled
    Checksumming of packets disabled
  Tunnel TTL 255
  Fast tunneling enabled
  Tunnel transmit bandwidth 8000 (kbps)
  Tunnel receive bandwidth 8000 (kbps)
  Tunnel protection via IPSec (profile "GRE-PROFILE")
  Last input 00:00:04, output never, output hang never
  Last clearing of "show interface" counters never
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: fifo
  Output queue: 0/0 (size/max)
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
```

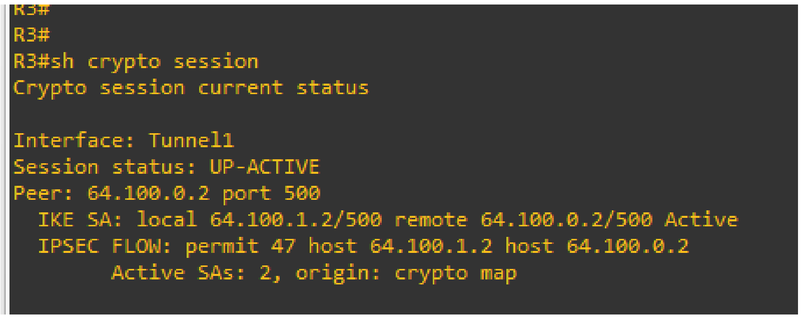
Step 8: On R1 and R3, verify the crypto settings.

On R1, use the show crypto session command to verify the operation of the VPN tunnel.



```
[OK]
R1#sh crypto session
Crypto session current status

Interface: FastEthernet0/0
Session status: UP-ACTIVE
Peer: 64.100.1.2 port 500
  IKE SA: local 64.100.0.2/500 remote 64.100.1.2/500 Active
  IPSEC FLOW: permit 47 host 64.100.0.2 host 64.100.1.2
        Active SAs: 2, origin: crypto map

R1#
R1#
```

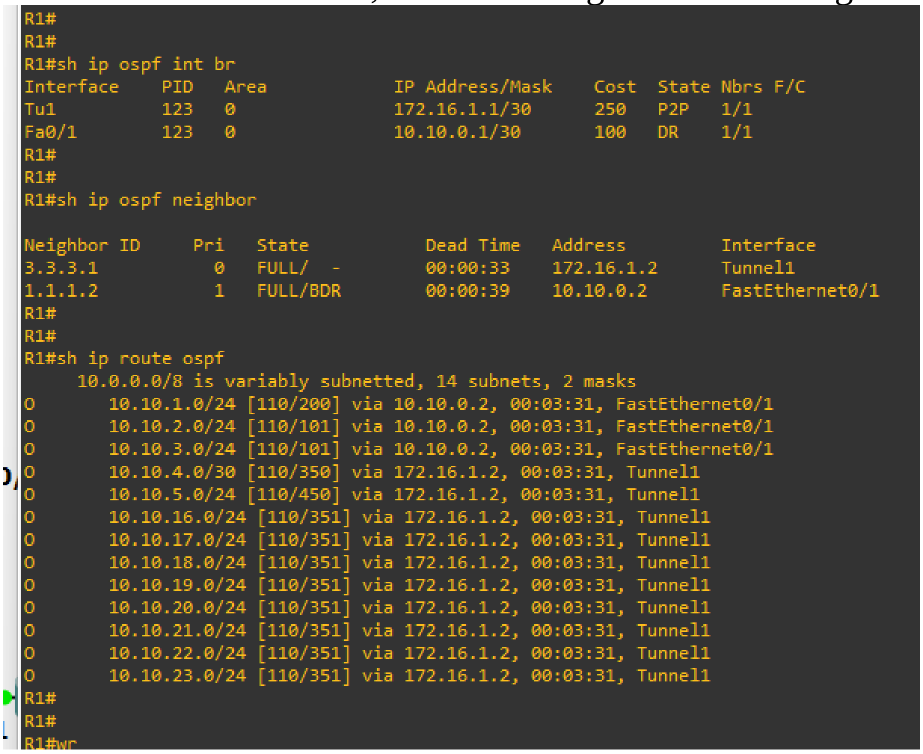
On R3, use the show crypto session command to verify the operation of the VPN tunnel.



```
R3#
R3#
R3#sh crypto session
Crypto session current status

Interface: Tunnel1
Session status: UP-ACTIVE
Peer: 64.100.0.2 port 500
  IKE SA: local 64.100.1.2/500 remote 64.100.0.2/500 Active
  IPSEC FLOW: permit 47 host 64.100.1.2 host 64.100.0.2
        Active SAs: 2, origin: crypto map
```

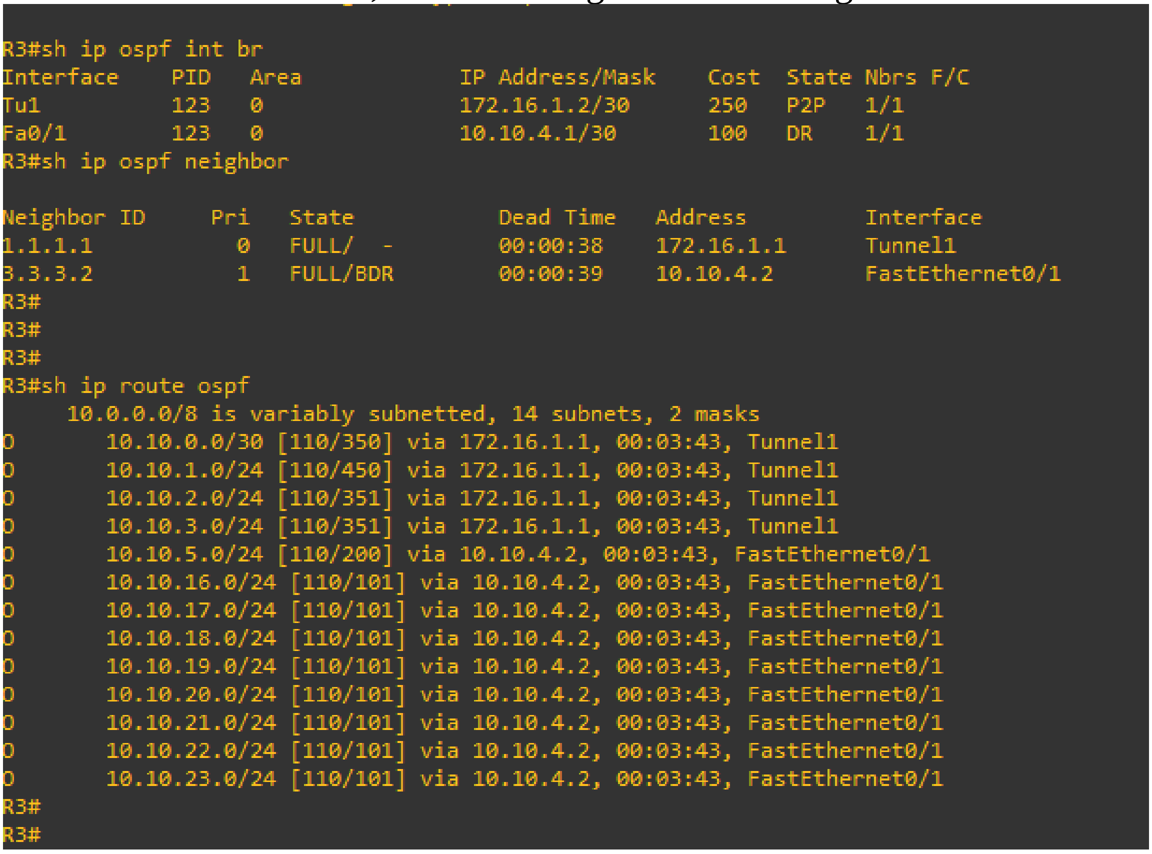
Step 9: On R1 and R3, verify OSPF routing.

On R1 verify which interfaces are configured for OSPF using the show ip ospf interface brief command, check the neighbor and routing table of OSPF.



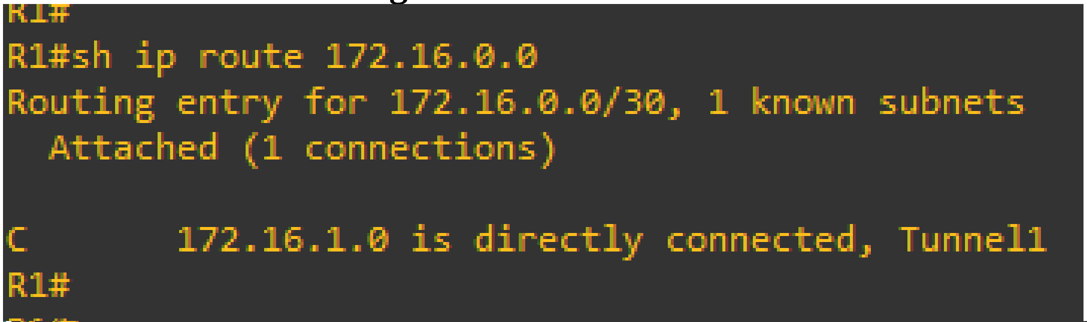
```
R1#
R1#
R1#sh ip ospf int br
Interface    PID   Area            IP Address/Mask    Cost  State Nbrs F/C
Tu1          123   0               172.16.1.1/30      250   P2P   1/1
Fa0/1        123   0               10.10.0.1/30       100   DR    1/1
R1#
R1#
R1#sh ip ospf neighbor

Neighbor ID     Pri   State           Dead Time   Address         Interface
3.3.3.1           0   FULL/  -        00:00:33    172.16.1.2      Tunnel1
1.1.1.2           1   FULL/BDR        00:00:39    10.10.0.2       FastEthernet0/1
R1#
R1#
R1#sh ip route ospf
     10.0.0.0/8 is variably subnetted, 14 subnets, 2 masks
O       10.10.1.0/24 [110/200] via 10.10.0.2, 00:03:31, FastEthernet0/1
O       10.10.2.0/24 [110/101] via 10.10.0.2, 00:03:31, FastEthernet0/1
O       10.10.3.0/24 [110/101] via 10.10.0.2, 00:03:31, FastEthernet0/1
O       10.10.4.0/30 [110/350] via 172.16.1.2, 00:03:31, Tunnel1
O       10.10.5.0/24 [110/450] via 172.16.1.2, 00:03:31, Tunnel1
O       10.10.16.0/24 [110/351] via 172.16.1.2, 00:03:31, Tunnel1
O       10.10.17.0/24 [110/351] via 172.16.1.2, 00:03:31, Tunnel1
O       10.10.18.0/24 [110/351] via 172.16.1.2, 00:03:31, Tunnel1
O       10.10.19.0/24 [110/351] via 172.16.1.2, 00:03:31, Tunnel1
O       10.10.20.0/24 [110/351] via 172.16.1.2, 00:03:31, Tunnel1
O       10.10.21.0/24 [110/351] via 172.16.1.2, 00:03:31, Tunnel1
O       10.10.22.0/24 [110/351] via 172.16.1.2, 00:03:31, Tunnel1
O       10.10.23.0/24 [110/351] via 172.16.1.2, 00:03:31, Tunnel1
R1#
R1#
R1#wr
```

On R1 verify which interfaces are configured for OSPF using the show ip ospf interface brief command, check the neighbor and routing table of OSPF.

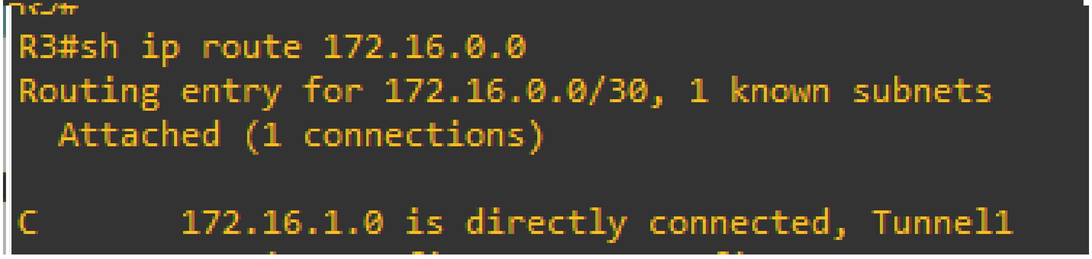


```
R3#sh ip ospf int br
Interface    PID   Area            IP Address/Mask    Cost  State Nbrs F/C
Tu1          123   0               172.16.1.2/30      250   P2P   1/1
Fa0/1        123   0               10.10.4.1/30       100   DR    1/1
R3#sh ip ospf neighbor

Neighbor ID     Pri   State           Dead Time   Address         Interface
1.1.1.1           0   FULL/  -        00:00:38    172.16.1.1      Tunnel1
3.3.3.2           1   FULL/BDR        00:00:39    10.10.4.2       FastEthernet0/1
R3#
R3#
R3#
R3#sh ip route ospf
     10.0.0.0/8 is variably subnetted, 14 subnets, 2 masks
O       10.10.0.0/30 [110/350] via 172.16.1.1, 00:03:43, Tunnel1
O       10.10.1.0/24 [110/450] via 172.16.1.1, 00:03:43, Tunnel1
O       10.10.2.0/24 [110/351] via 172.16.1.1, 00:03:43, Tunnel1
O       10.10.3.0/24 [110/351] via 172.16.1.1, 00:03:43, Tunnel1
O       10.10.5.0/24 [110/200] via 10.10.4.2, 00:03:43, FastEthernet0/1
O       10.10.16.0/24 [110/101] via 10.10.4.2, 00:03:43, FastEthernet0/1
O       10.10.17.0/24 [110/101] via 10.10.4.2, 00:03:43, FastEthernet0/1
O       10.10.18.0/24 [110/101] via 10.10.4.2, 00:03:43, FastEthernet0/1
O       10.10.19.0/24 [110/101] via 10.10.4.2, 00:03:43, FastEthernet0/1
O       10.10.20.0/24 [110/101] via 10.10.4.2, 00:03:43, FastEthernet0/1
O       10.10.21.0/24 [110/101] via 10.10.4.2, 00:03:43, FastEthernet0/1
O       10.10.22.0/24 [110/101] via 10.10.4.2, 00:03:43, FastEthernet0/1
O       10.10.23.0/24 [110/101] via 10.10.4.2, 00:03:43, FastEthernet0/1
R3#
R3#
```

Step 10: Verify that there is an operational logical point-to-point link between R1 and R3 using the GRE tunnel interface.



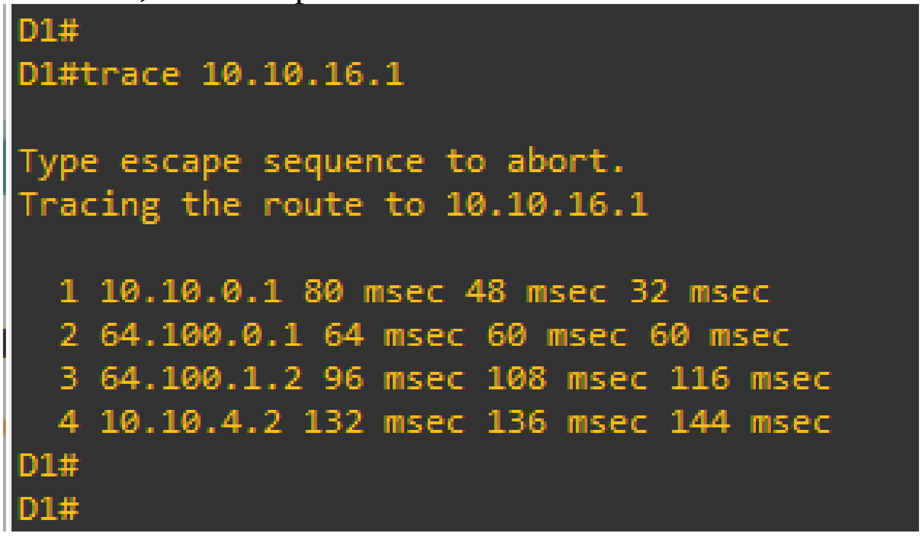
```
R1#
R1#sh ip route 172.16.0.0
Routing entry for 172.16.0.0/30, 1 known subnets
  Attached (1 connections)

C       172.16.1.0 is directly connected, Tunnel1
R1#
```



```
R3#
R3#sh ip route 172.16.0.0
Routing entry for 172.16.0.0/30, 1 known subnets
  Attached (1 connections)

C       172.16.1.0 is directly connected, Tunnel1
```

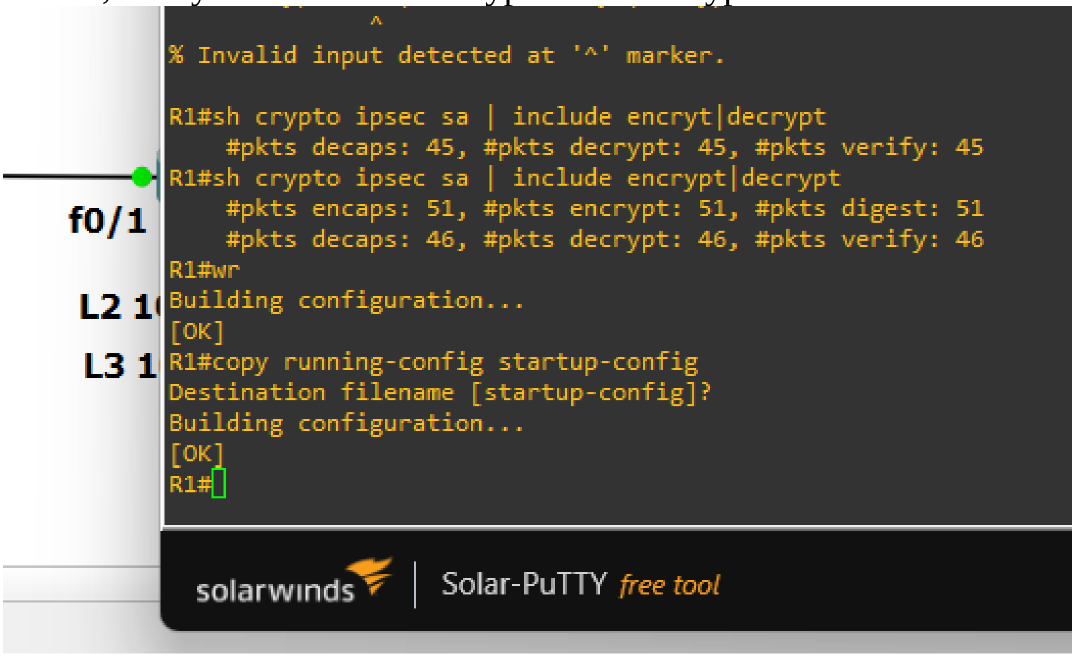
Step 11: Test the GRE over IPsec VPN tunnel.

From D1, trace the path taken to the R3 10.10.16.1 interface.



```
D1#
D1#trace 10.10.16.1

Type escape sequence to abort.
Tracing the route to 10.10.16.1

  1 10.10.0.1 80 msec 48 msec 32 msec
  2 64.100.0.1 64 msec 60 msec 60 msec
  3 64.100.1.2 96 msec 108 msec 116 msec
  4 10.10.4.2 132 msec 136 msec 144 msec
D1#
D1#
```

On R1, verify the IPsec SA encrypted and decrypted statistics.



```
               ^
% Invalid input detected at '^' marker.

R1#sh crypto ipsec sa | include encryt|decrypt
    #pkts decaps: 45, #pkts decrypt: 45, #pkts verify: 45
R1#sh crypto ipsec sa | include encrypt|decrypt
    #pkts encaps: 51, #pkts encrypt: 51, #pkts digest: 51
    #pkts decaps: 46, #pkts decrypt: 46, #pkts verify: 46
R1#wr
Building configuration...
[OK]
R1#copy running-config startup-config
Destination filename [startup-config]?
Building configuration...
[OK]
R1#
```

The output verifies that the GRE over IPsec VPN tunnel is properly encrypting traffic between both sites. The packets encrypted include the trace packets along with OSPF packets.