PRACTICAL 4

MIHIR DEVGHARE (20162171006)
NETWORK SECURITY

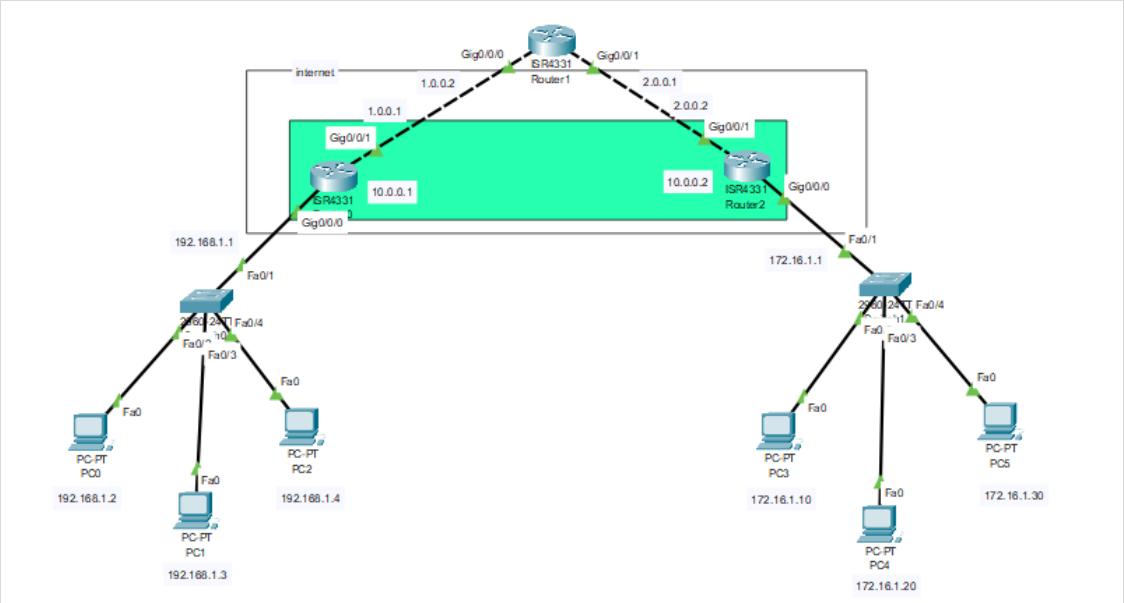
The requirement is to create a tunnel between Site A and Site B, which is located at 2 different geographical location. The tunnel protocol you can use GRE and analyse the packet that is travelling through the tunnel.

Ensure that there are multiple routers present in the internet zone. And still the tunnel should work even if the site A and site B network details are not present with the routers present in the internet zone.

Kindly share the configuration details along with the network topology.

Tunnel using 3 routers:

Create the topology as shown below:



Above, assign the IP addresses and gateway as well.

Note: IP addresses 10.0.0.1 and 10.0.0.2 are private IP addresses for tunnels and they are not assigned yet. Also, don't set the Routing table as we are creating a virtual network so that the third party router (router 1) doesn't know anything about router 0 and router 2.

Now, open the terminal of routers and there we will configure the virtual private network.

Router 0:

```
Router(config)#ip route 0.0.0.0 0.0.0.0 1.0.0.2
Router(config)#interface tunnel 10

Router(config-if)#
%LINK-5-CHANGED: Interface Tunnel10, changed state to up

Router(config-if)#ip address 10.0.0.1 255.0.0.0
Router(config-if)#tunnel destination 2.0.0.2
Router(config-if)#tunnel source hih
Router(config-if)#tunnel source gig
Router(config-if)#tunnel source gigabitEthernet 0/0/1
Router(config-if)#
%LINEPROTO-5-UPDOWN: Line protocol on Interface Tunnel10, changed state to up

Router(config-if)#tunnel mode gre ip
Router(config-if)#
```

Above, **ip route 0.0.0.0 0.0.0.0 1.0.0.2** says that any data/packet passing through the router will have to go through the router having IP address 1.0.0.2.

ip address 10.0.0.1 255.0.0.0 is the ip address and subnet mask of the tunnel which we are creating via router 0.

Rest, we have provided the required things such as tunnel number, tunnel destination, source cable, and then set the tunnel mode as gre.

Router 2:

```
Router#conf
Router#configure t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#ip route 0.0.0.0 0.0.0.0 2.0.0.1
Router(config)#interface tunnel 10

Router(config-if)#
%LINK-5-CHANGED: Interface Tunnel10, changed state to up

Router(config-if)#ip address 10.0.0.2 255.0.0.0
Router(config-if)#tunnel des
Router(config-if)#tunnel destination 1.0.0.1
Router(config-if)#tunnel source gig
Router(config-if)#tunnel source gigabitEthernet 0/0/1
Router(config-if)#
%LINEPROTO-5-UPDOWN: Line protocol on Interface Tunnel10, changed state to up

Router(config-if)#tunnel mode gre ip
Router(config-if)#
```

ip address 10.0.0.2 255.0.0.0 is the ip address and subnet mask of the tunnel which we are creating via router 0.

Rest, we have provided the required things such as tunnel number, tunnel destination, source cable, and then set the tunnel mode as gre.

The configuration part has been done. Checking if the configuration is successful or not:

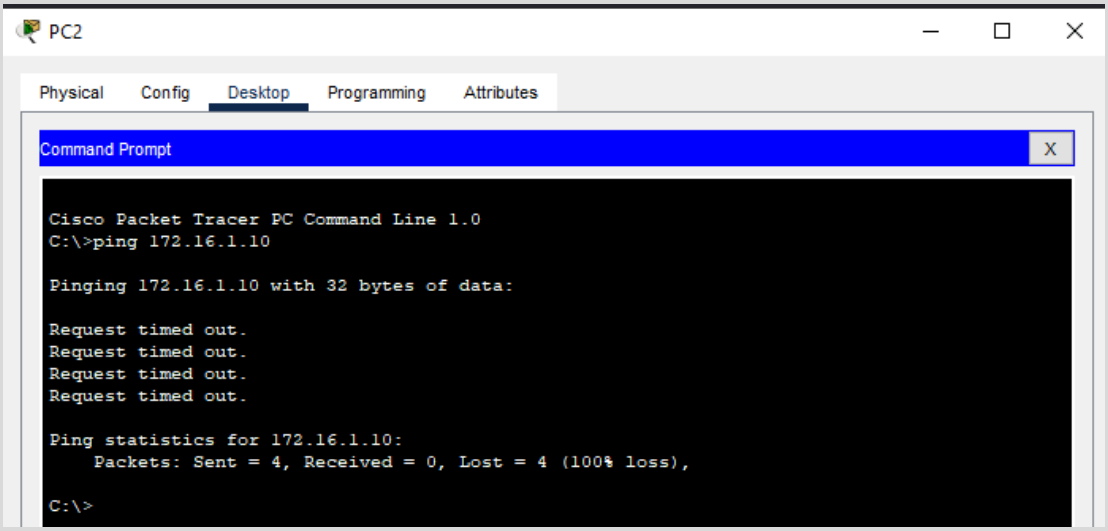
Router 0:

```
Router#ping 2.0.0.2

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2.0.0.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 0/0/4 ms

Router#
```

Router 2:

```
Router#ping 1.0.0.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 1.0.0.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 0/0/1 ms

Router#
```

Now, the routers are able to communicate with each other. But the path setting via tunnel is remaining and because of that, the end devices are still not able to communicate with each other.



So, now we will set the path for these devices so that they can communicate with each other.

Router 0:

```
Router(config)#ip route 172.16.1.0 255.255.255.0 10.0.0.2
Router(config)#
```

Router 2:

```
Router(config)#ip route 192.168.1.0 255.255.255.0 10.0.0.1
Router(config)#
```

Now, we will try to ping again the same ip address from the same pc (PC2):

```
C:\>ping 172.16.1.10

Pinging 172.16.1.10 with 32 bytes of data:

Reply from 172.16.1.10: bytes=32 time<1ms TTL=126
Reply from 172.16.1.10: bytes=32 time<1ms TTL=126
Reply from 172.16.1.10: bytes=32 time<1ms TTL=126
Reply from 172.16.1.10: bytes=32 time<1ms TTL=126

Ping statistics for 172.16.1.10:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\>
```

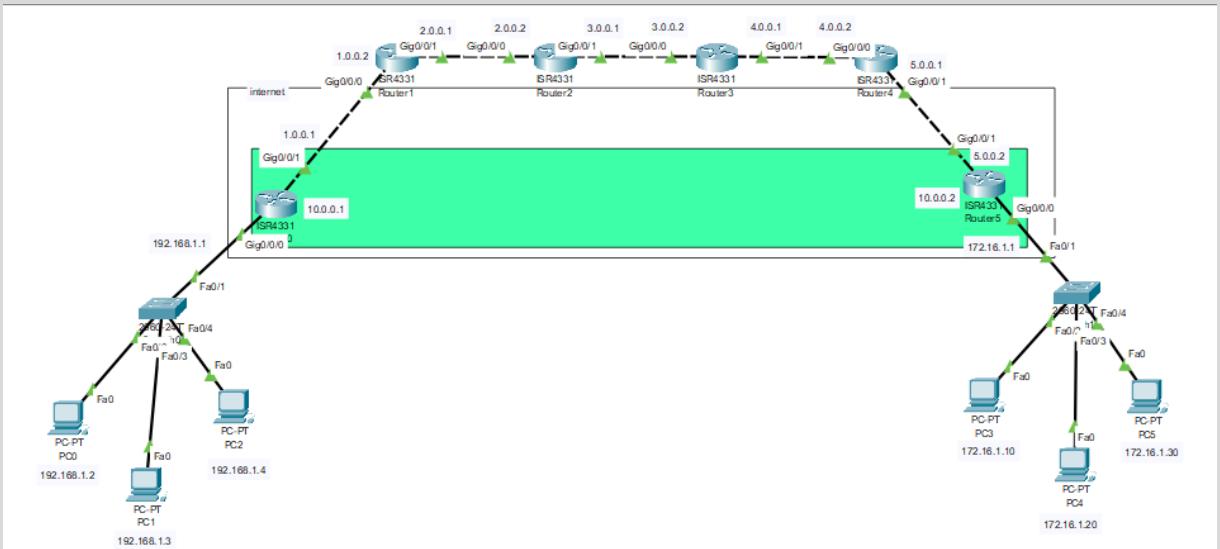
It is pinging, but to confirm that it is communicating through the tunnel, we will perform the tracert command:

```
C:\>tracert 172.16.1.10

Tracing route to 172.16.1.10 over a maximum of 30 hops:

  1    0 ms      0 ms      0 ms      192.168.1.1
  2    0 ms      0 ms      0 ms      10.0.0.2
  3    0 ms      0 ms      1 ms      172.16.1.10

Trace complete.

C:\>
```

We can see above that the IP address 10.0.0.2 is there which is the IP address of the tunnel, hence the communication is being done via the private tunnel.

Tunnel using 6 routers:

Create the topology as shown below:



Above, assign the IP addresses and gateway as well.

Note: IP addresses 10.0.0.1 and 10.0.0.2 are private IP addresses for tunnels and they are not assigned yet. Also, don’t set the Routing table as we are creating a virtual network so that the third party routers (router 1...4) doesn’t know anything about router 0 and router 5.

Now, open the terminal of routers and there we will configure the virtual private network.

Router 0:

```
Router>en
Router#config t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#ip route 0.0.0.0 0.0.0.0 1.0.0.2
Router(config)#interface tunnel 10

Router(config-if)#
%LINK-5-CHANGED: Interface Tunnel10, changed state to up

Router(config-if)#ip address 10.0.0.1 255.0.0.0
Router(config-if)#tunnel destina
Router(config-if)#tunnel destination 5.0.0.2
Router(config-if)#tunnel source gig
Router(config-if)#tunnel source gigabitEthernet 0/0/1
Router(config-if)#
%LINEPROTO-5-UPDOWN: Line protocol on Interface Tunnel10, changed state to up

Router(config-if)#tunnel mode gre ip
Router(config-if)#
```

Above, **ip route 0.0.0.0 0.0.0.0 1.0.0.2** says that any data/packet passing through the router will have to go through the router having IP address 1.0.0.2.

ip address 10.0.0.1 255.0.0.0 is the ip address and subnet mask of the tunnel which we are creating via router 0.

Rest, we have provided the required things such as tunnel number, tunnel destination, source cable, and then set the tunnel mode as gre.

Router 5:

```
Router>en
Router#config t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#ip route 0.0.0.0 0.0.0.0 5.0.0.1
Router(config)#interface tunnel 10

Router(config-if)#
%LINK-5-CHANGED: Interface Tunnel10, changed state to up

Router(config-if)#ip address 10.0.0.2 255.0.0.0
Router(config-if)#tunnel destination 1.0.0.1
Router(config-if)#tunnel source gig
Router(config-if)#tunnel source gigabitEthernet 0/0/1
Router(config-if)#
%LINEPROTO-5-UPDOWN: Line protocol on Interface Tunnel10, changed state to up

Router(config-if)#tunnel mode gre ip
Router(config-if)#
```

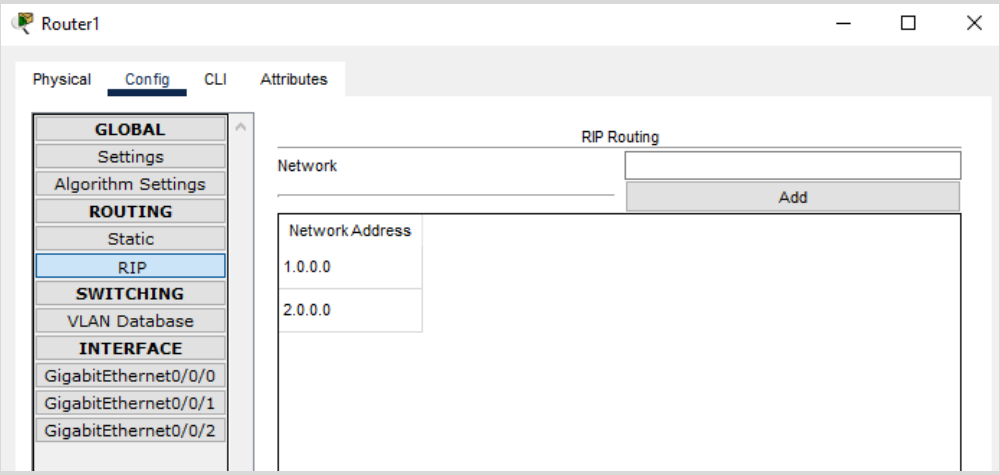
ip address 10.0.0.2 255.0.0.0 is the ip address and subnet mask of the tunnel which we are creating via router 0.

Rest, we have provided the required things such as tunnel number, tunnel destination, source cable, and then set the tunnel mode as gre.

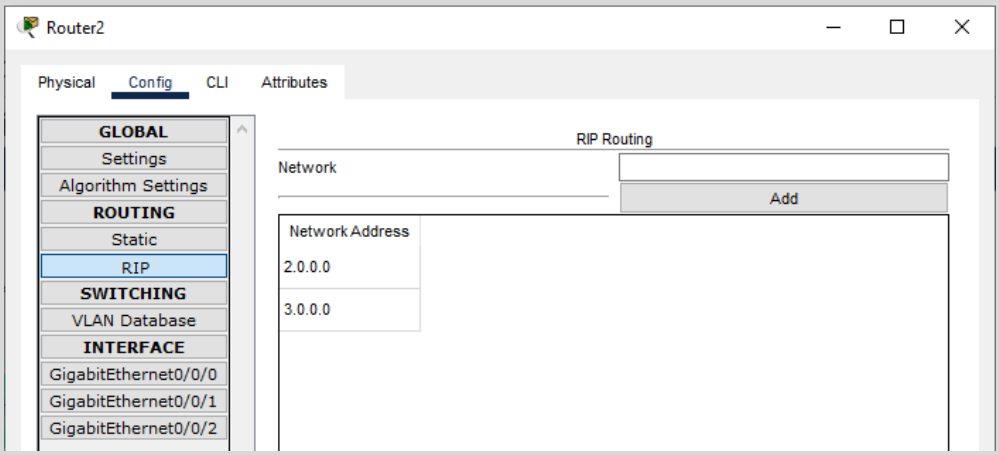
The configuration part has been done yet we will not be able to send the messages or ping the routers as we have not set the routing table as there are multiple routers in this organization.

Setting the routing table via RIP:

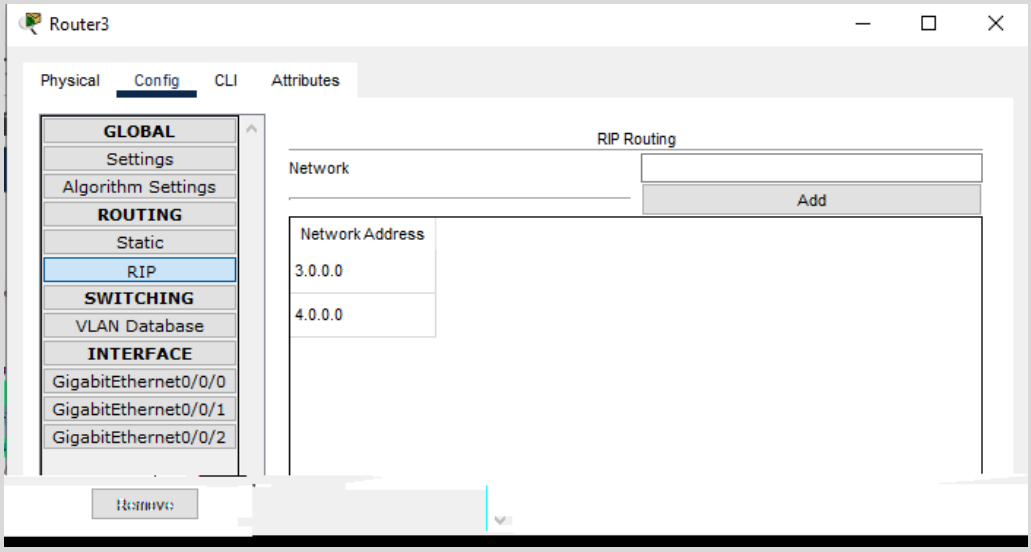
Router 1:



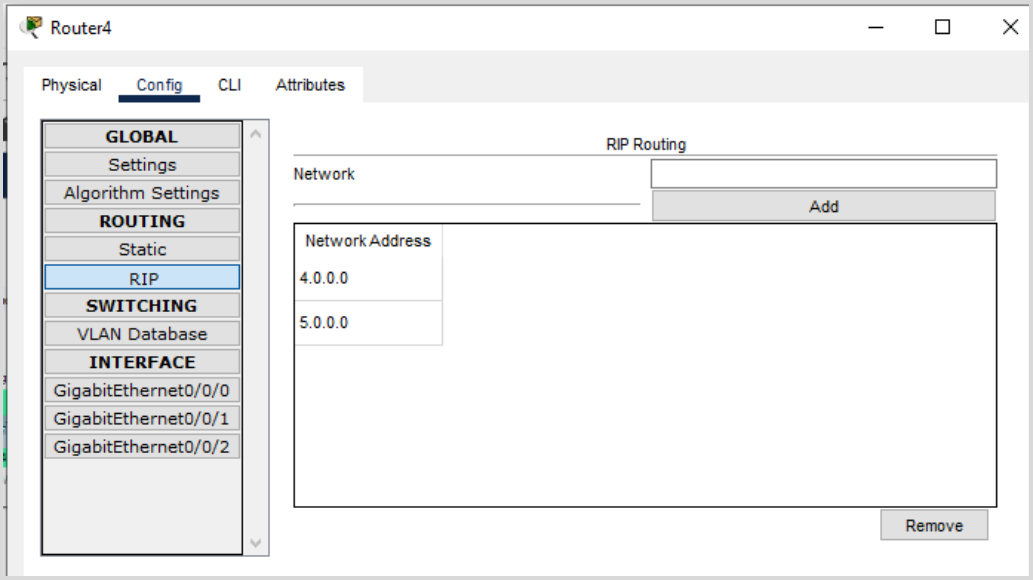
Router 2:



Router 3:



Router 4:



The configuration part has been done. Checking if the configuration is successful or not:

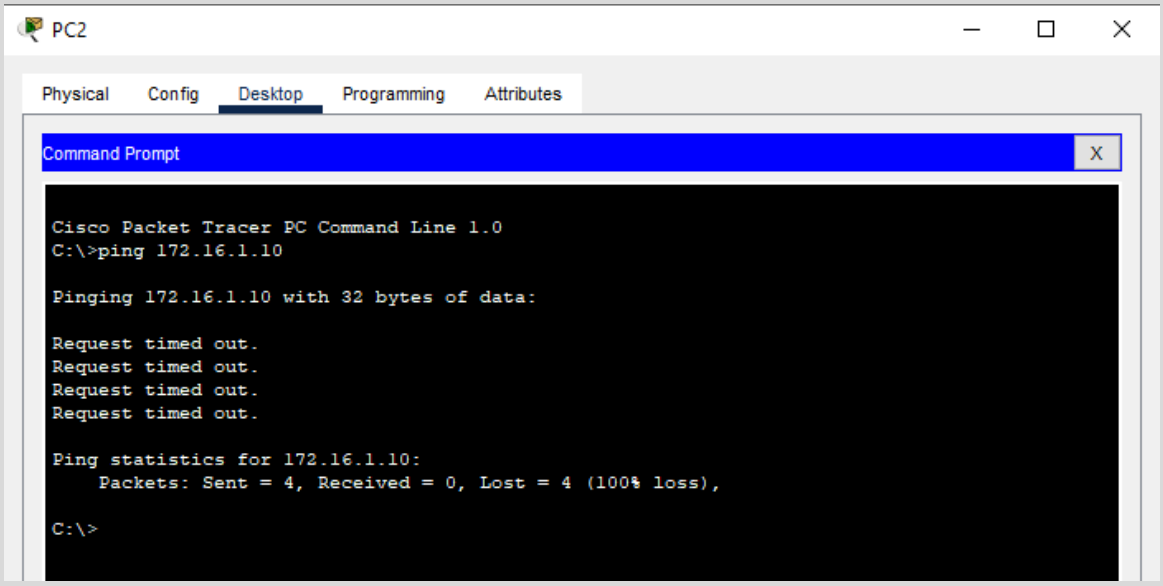
Router 0:

```
Router#ping 5.0.0.2

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 5.0.0.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 0/0/0 ms

Router#
```

Router 5:

```
Router#ping 1.0.0.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 1.0.0.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 0/0/1 ms

Router#
```

Now, the routers are able to communicate with each other. But the path setting via tunnel is remaining and because of that, the end devices are still not able to communicate with each other.



So, now we will set the path for these devices so that they can communicate with each other.

Router 0:

```
Router(config)#ip route 172.16.1.0 255.255.255.0 10.0.0.2
Router(config)#
```

Router 5:

```
Router(config)#ip route 192.168.1.0 255.255.255.0 10.0.0.1
Router(config)#
```

Now, we will try to ping again the same ip address from the same pc (PC2):

```
C:\>ping 172.16.1.10

Pinging 172.16.1.10 with 32 bytes of data:

Reply from 172.16.1.10: bytes=32 time<1ms TTL=126
Reply from 172.16.1.10: bytes=32 time=11ms TTL=126
Reply from 172.16.1.10: bytes=32 time=11ms TTL=126
Reply from 172.16.1.10: bytes=32 time=31ms TTL=126

Ping statistics for 172.16.1.10:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 31ms, Average = 13ms

C:\>
```

It is pinging, but to confirm that it is communicating through the tunnel, we will perform the tracert command:

```
C:\>tracert 172.16.1.10

Tracing route to 172.16.1.10 over a maximum of 30 hops:

  1    0 ms      0 ms      0 ms      192.168.1.1
  2   11 ms     10 ms     10 ms      10.0.0.2
  3   10 ms     11 ms     11 ms      172.16.1.10

Trace complete.

C:\>
```

We can see above that the IP address 10.0.0.2 is there which is the IP address of the tunnel, hence the communication is being done via the private tunnel.