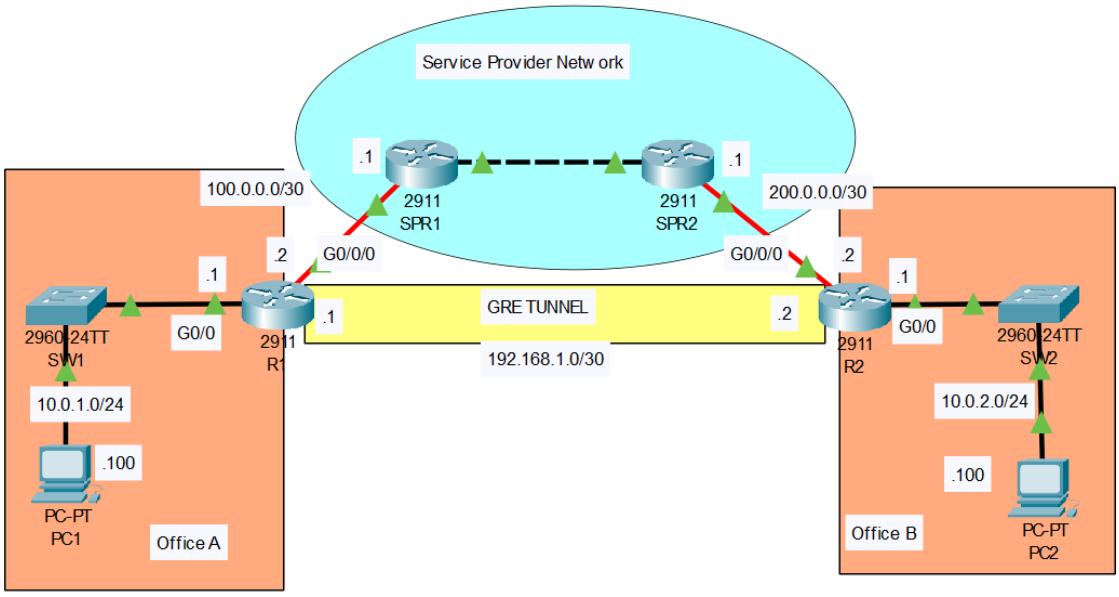
Student: Mono

Lab: Jeremy IT's Lab

Topic: GRE



1. Configure a GRE tunnel to connect R1 and R2.

2. Configure OSPF on the tunnel interfaces of R1 and R2, to allow PC1 and PC2 to communicate.

1.

Int Tunnel

Tunnel source

Tunnel destination

Int Tunnel ip address itself for the tunnel

R1

```
R1>
R1>en
R1#config t
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#int ?
  Dialer            Dialer interface
  Dot11Radio        Dot11 interface
  Ethernet          IEEE 802.3
  FastEthernet      FastEthernet IEEE 802.3
  GigabitEthernet   GigabitEthernet IEEE 802.3z
  Loopback          Loopback interface
  Port-channel      Ethernet Channel of interfaces
  Serial            Serial
  Tunnel            Tunnel interface
  Virtual-Template  Virtual Template interface
  Vlan              Catalyst Vlans
  range             interface range command
R1(config)#int tunnel ?
  <0-2147483647>  Tunnel interface number
R1(config)#int tunnel 0

R1(config-if)#
%LINK-5-CHANGED: Interface Tunnel0, changed state to up

R1(config-if)#tunnel ?
  destination  destination of tunnel
  mode         tunnel encapsulation method
  source       source of tunnel packets
R1(config-if)#tunnel source ?
  Ethernet         IEEE 802.3
  FastEthernet     FastEthernet IEEE 802.3
  GigabitEthernet  GigabitEthernet IEEE 802.3z
  Loopback         Loopback interface
  Serial           Serial
R1(config-if)#tunnel source g0/0/0
R1(config-if)#tunnel destination ?
  A.B.C.D  ip address
R1(config-if)#tunnel destination 200.0.0.2
R1(config-if)#ip add 192.168.1.1 255.255.255.252
R1(config-if)#
```

Verify – R1

```
R1(config)#do show ip int bri
Interface              IP-Address      OK? Method Status                Protocol
GigabitEthernet0/0     10.0.1.1        YES NVRAM  up                    up
GigabitEthernet0/1     unassigned      YES NVRAM  administratively down down
GigabitEthernet0/2     unassigned      YES NVRAM  administratively down down
GigabitEthernet0/0/0   100.0.0.2       YES manual up                    up
Tunnel0                192.168.1.1     YES manual up                    down
Vlan1                  unassigned      YES unset  administratively down down
```

Note: The protocol of tunnel 0 is down!

R2 – Config like R1

```
R2>en
R2#config t
Enter configuration commands, one per line.  End with CNTL/Z.
R2(config)#int tunnel 0

R2(config-if)#
%LINK-5-CHANGED: Interface Tunnel0, changed state to up

R2(config-if)#tunnel source g0/0/0
R2(config-if)#tunnel destination 100.0.0.2
R2(config-if)#
R2(config-if)#ip address 192.168.1.2 255.255.255.252
R2(config-if)#
R2(config-if)#do show ip int bri
Interface              IP-Address      OK? Method Status                Protocol
GigabitEthernet0/0     10.0.2.1        YES NVRAM  up                    up
GigabitEthernet0/1     unassigned      YES NVRAM  administratively down down
GigabitEthernet0/2     unassigned      YES NVRAM  administratively down down
GigabitEthernet0/0/0   200.0.0.2       YES manual up                    up
Tunnel0                192.168.1.2     YES manual up                    down
Vlan1                  unassigned      YES unset  administratively down down
R2(config-if)#
```

After checking both ip routing tables of R1 and R2, there are no routes for them to reach each other. Therefore, configuring default routes for them is necessary!

R1

```
     10.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C       10.0.1.0/24 is directly connected, GigabitEthernet0/0
L       10.0.1.1/32 is directly connected, GigabitEthernet0/0
     100.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C       100.0.0.0/30 is directly connected, GigabitEthernet0/0/0
L       100.0.0.2/32 is directly connected, GigabitEthernet0/0/0
```

R2

```
     10.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C       10.0.2.0/24 is directly connected, GigabitEthernet0/0
L       10.0.2.1/32 is directly connected, GigabitEthernet0/0
     200.0.0.0/24 is variably subnetted, 2 subnets, 2 masks
C       200.0.0.0/30 is directly connected, GigabitEthernet0/0/0
L       200.0.0.2/32 is directly connected, GigabitEthernet0/0/0
```

Config default routing!

```
R1(config)#ip route 0.0.0.0 0.0.0.0 100.0.0.1

R2(config)#ip route 0.0.0.0 0.0.0.0 200.0.0.1
```

Verify

R1

```
     10.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C       10.0.1.0/24 is directly connected, GigabitEthernet0/0
L       10.0.1.1/32 is directly connected, GigabitEthernet0/0
     100.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C       100.0.0.0/30 is directly connected, GigabitEthernet0/0/0
L       100.0.0.2/32 is directly connected, GigabitEthernet0/0/0
     192.168.1.0/24 is variably subnetted, 2 subnets, 2 masks
C       192.168.1.0/30 is directly connected, Tunnel0
L       192.168.1.1/32 is directly connected, Tunnel0
S*   0.0.0.0/0 [1/0] via 100.0.0.1
```

R2

```
     10.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C       10.0.2.0/24 is directly connected, GigabitEthernet0/0
L       10.0.2.1/32 is directly connected, GigabitEthernet0/0
     192.168.1.0/24 is variably subnetted, 2 subnets, 2 masks
C       192.168.1.0/30 is directly connected, Tunnel0
L       192.168.1.2/32 is directly connected, Tunnel0
     200.0.0.0/24 is variably subnetted, 2 subnets, 2 masks
C       200.0.0.0/30 is directly connected, GigabitEthernet0/0/0
L       200.0.0.2/32 is directly connected, GigabitEthernet0/0/0
S*   0.0.0.0/0 [1/0] via 200.0.0.1
```

After I configured the default routes for R1 and R2, the routes of the tunnels showed up!

Now, R1 and R2 can ping each other using ip addresses of the tunnels.

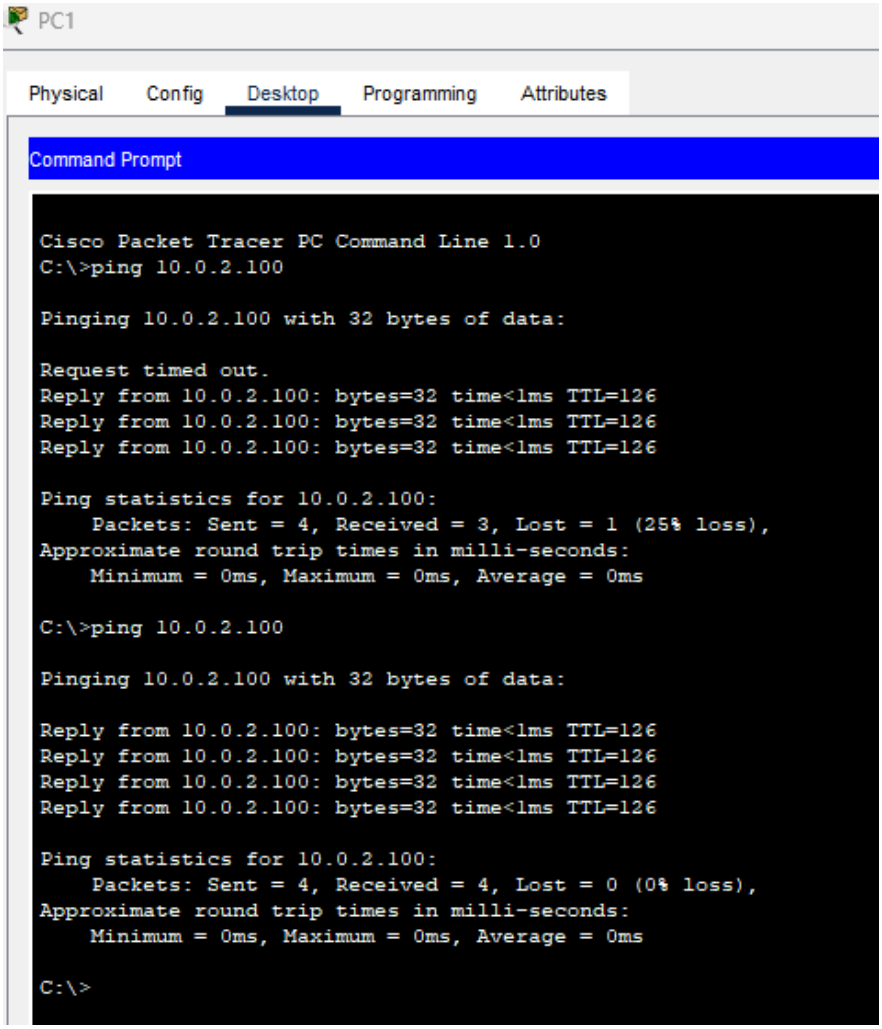
2.

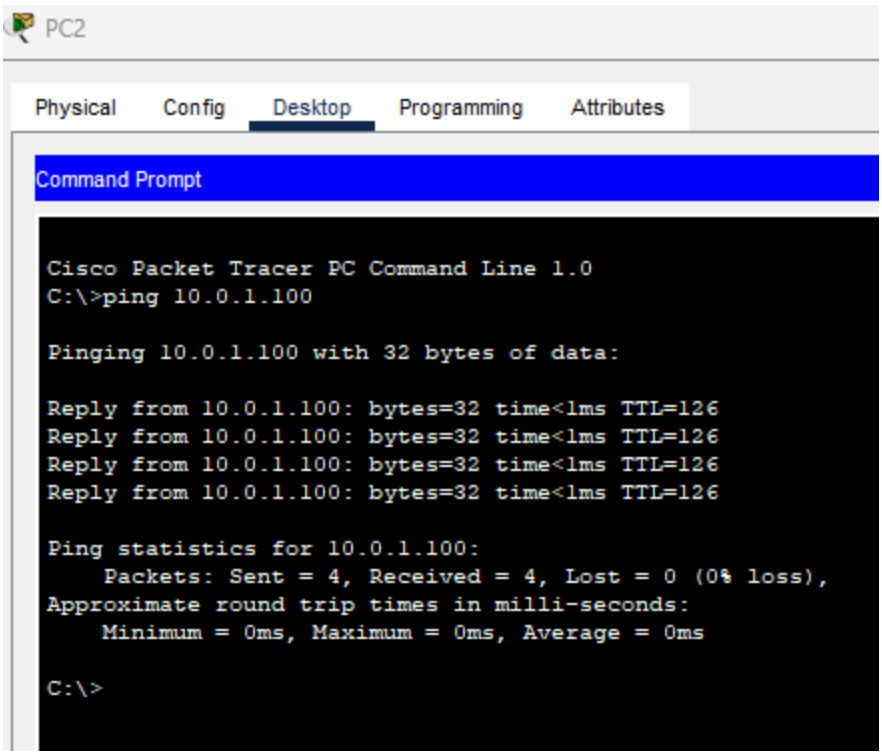
However, PC1 cannot ping PC2, vice versa because there are no routes to reach from the PCs' networks. That is why I need to configure OSPF on the tunnel interfaces of R1 and R2.

```
R1>en
R1#config t
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#router ospf 1
R1(config-router)#network 192.168.1.1 0.0.0.0 area 0
R1(config-router)#network 10.0.1.1 0.0.0.0 area 0
R1(config-router)#passive int g0/0
                           ^
% Invalid input detected at '^' marker.

R1(config-router)#passive-int g0/0
R1(config-router)#
```

```
R2>en
R2#config t
Enter configuration commands, one per line.  End with CNTL/Z.
R2(config)#router ospf 1
R2(config-router)#network ?
  A.B.C.D  Network number
R2(config-router)#network 192.168.1.2 ?
  A.B.C.D  OSPF wild card bits
R2(config-router)#network 192.168.1.2 0.0.0.0 ?
  area  Set the OSPF area ID
R2(config-router)#network 192.168.1.2 0.0.0.0 area 0

R2(config-router)#network 10.0.2.1 0.0.0.0 area 0
R2(config-router)#pass
R2(config-router)#passive-interface g0/0
R2(config-router)#exit
R2(config)#
R2(config)#exit
```

Verify





Another option, I can configure specific static routes for PC1 and PC2 reaching each other.