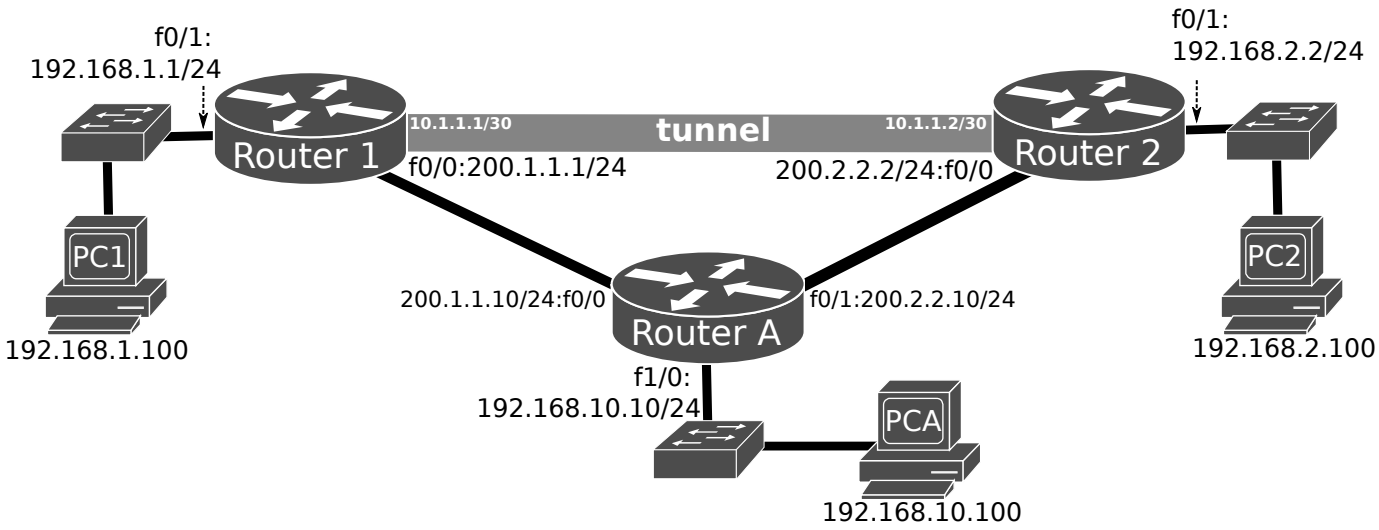
Redes de Comunicações 2

IPv4 Tunnels



1.1. Assemble the above depicted network, start by configuring all interfaces' IPv4 addresses and OSPF in all routers and respective interfaces. Verify the interfaces' configurations, IPv4 routing table, and that devices have fully connectivity.

1.2. Configure an IPv4-IPv4 tunnel between Router1 and Router2 (as depicted in figure):

```
Router1(config)# interface Tunnel 0                    !Tunnels can be numbered from 0 to 2147483647
Router1(config-if)# tunnel source 200.1.1.1
Router1(config-if)# tunnel destination 200.2.2.2
Router1(config-if)# tunnel mode ipip
...
Router2(config)# interface Tunnel 0
Router2(config-if)# tunnel source 200.2.2.2
Router2(config-if)# tunnel destination 200.1.1.1
Router2(config-if)# tunnel mode ipip
```

Check the status of Tunnel 0 on both routers:

```
show interface Tunnel 0
```

Configure a static route from Router1 to network 192.168.2.0/24 via Tunnel 0 (via destination IP):

```
Router1(config)# ip route 192.168.2.0 255.255.255.0 Tunnel 0
```

Verify the routing table.

>> Why is not the static route active in the routing table?

Note: tunnel end points virtual interfaces (VTI) do not require the same ID number, however, good configuration guidelines strongly recommend it.

1.3. Associate the network 10.1.1.0/30 to the Tunnel and configure the end-points IPv4 addresses:

```
Router1(config)# interface Tunnel 0
Router1(config-if)# ip address 10.1.1.1 255.255.255.252
...
Router3(config)# interface Tunnel 0
Router3(config-if)# ip address 10.1.1.2 255.255.255.252
```

>> Verify the routing table (if the static route is active).

>> What do you conclude about the requirement to assigned an IP network to the tunnel connection?

Note: Is not recommend, however, Cisco's IOS also allows to "map" the IP address of a physical interface with the command: ip unnumbered f0/0.

1.4. Start a capture on network 200.1.1.0/24 and perform a ping from PC1 to PC2. Analyze the captured packets.

>> Analyze the tunneled packets (multiple IPv4 headers).

>> Explain the purpose of each IPv4 header.

1.5. Change the type of the Tunnel to GRE IPv4:

```
Router1(config)# interface Tunnel 0
Router1(config-if)# tunnel mode gre ip
```

Verify the routing table (if the static route is active), restart a capture on network 200.1.1.0/24 and perform a ping from PC1 to PC2.

>> Analyze the tunneled packets (multiple IPv4 headers and GRE header).

>> Explain the purpose of each IPv4 header and GRE header.

1.6. Configure the following static route at Router 2:

```
Router2(config)# ip route 192.168.10.0 255.255.255.0 tunnel 0
```

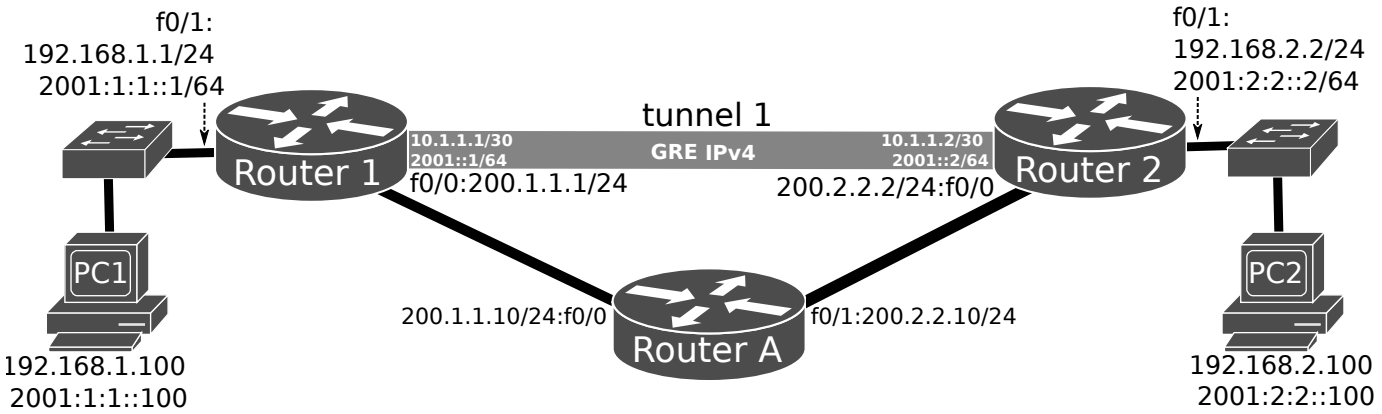
Verify the routing table (if the static route is active), restart a capture on network 200.1.1.0/24 and perform a ping from PC2 to PCA.

>> Analyze the tunneled packets (multiple IPv4 headers and GRE header).

>> Analyze the packets routed from Router 1 to Router A (duplicated ICMP packets).

>> Explain how a tunnel can be used to force routing via a node not on the “natural” path of traffic.

IPv6 over IPv4 Tunneling



2.1. Assuming that Router A does not support IPv6, add IPv6 addresses to the previous network.

Activate IPv6 routing: ipv6 unicast-routing

Test the IPv6 connectivity between the devices.

>> What do you conclude about the IPv6 connectivity?

2.2 Configure a manual IPv6 over IPv4 overlay tunnel between Router 1 and Router 2:

```
Router1(config)# interface Tunnel1
Router1(config-if)# ipv6 address 2001::1/64
Router1(config-if)# tunnel source f0/0
Router1(config-if)# tunnel destination 200.2.2.2
Router1(config-if)# tunnel mode ipv6ip
```

Repeat similar/symmetric configuration in Router2. Re-verify the routing tables and test the IPv6 connectivity between the devices.

>> What do you conclude about the IPv6 connectivity?

>> Explain the purpose of the configured tunnel.

2.3. Restart a capture on network 200.1.1.0/24. From Router1 ping Router2 Tunnel 1's IPv6 address, and vice-versa. Analyze the captured packets.

>> Analyze the tunneled packets (IPv4 and IPv6 headers).

>> Identify which header/protocol is handles the connection between Router1 and Router2.

>> Explain the purpose of each IPv4 and IPv6 header.

2.4. Restart a capture on network 200.1.1.0/24. Execute all necessary static routing configurations in order to obtain full connectivity:

```
Router1(config)# ipv6 route 2001:2:2::/64 2001::2
...
Router2(config)# ipv6 route 2001:1:1::/64 2001::1
```

From PC1 ping PC2 using IPv6. Analyze the captured packets.

>> Analyze the tunneled packets (IPv4 and IPv6 headers).

>> Identify which header/protocol is handles the connection between Router1 and Router2.

>> Explain the purpose of each IPv4 and IPv6 header.

2.5. Reconfigure in Router1 and Router2 the tunnel to GRE over IPv4 mode (to transport IPv4 and IPv6 traffic):

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

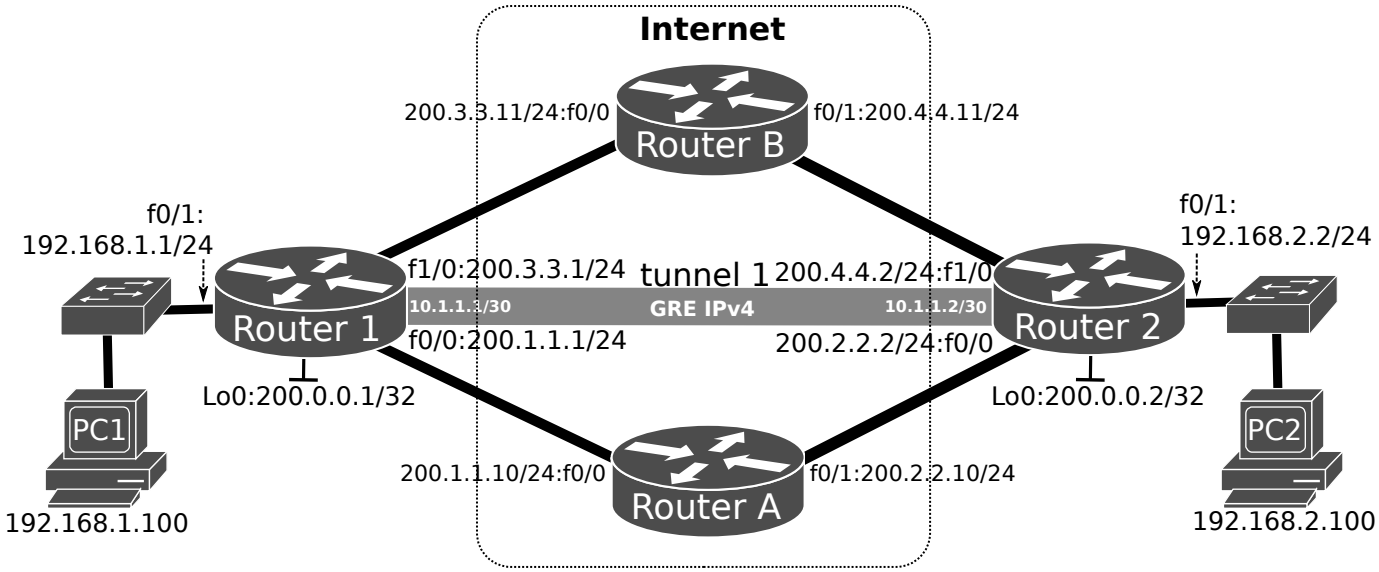
Restart a capture on network 200.1.1.0/24. Generate IPv4 and IPv6 traffic through the tunnel. Analyze the captured packets.

>> Analyze the tunneled packets (IPv4 and IPv6 headers and GRE header).

>> Explain the purpose of each IP header and GRE header.

(Optional) GRE IPv4 Tunnel / Overlay Network / Loopback Interfaces

In a realistic scenario, where NAT is not recommended, private corporate networks are not known by the Internet core routers and it cannot route traffic between them. It is required to create a virtual link (tunnel) between the private networks.



3.1. Configure the IPv4 addresses and activate OSPF for the underlying IPv4 network of the Internet core (including Loopback interfaces' addresses) in Routers 1, 2, A and B. To activate OSPF (process 1) in Router 1 for the underlying network:

```
Router1(config)# interface Loopback0
Router1(config-if)# ip ospf 1 area 0
Router1(config-if)# interface FastEthernet0/0
Router1(config-if)# ip ospf 1 area 0
Router1(config-if)# interface FastEthernet1/0
Router1(config-if)# ip ospf 1 area 0
```

Perform the equivalent configurations on Routers A, B and 2.

Verify the IPv4 routing tables (show ip route) and test the connectivity in the Internet core.

```
Router1# ping 200.0.0.2
Router1# ping 200.0.0.2 source Loopback 0
...
Router2# ping 200.0.0.1
Router2# ping 200.0.0.1 source Loopback 0
```

Test also the lack of connectivity between the private networks by ping PC2 from PC1 and vice-versa.

>> Explain why the private networks cannot be included in the Internet core routing domain.

>> Could they be included if the IPv4 networks add public IPv4 addresses?

3.2. Configure an GRE IPv4 tunnel (with overlay networks 10.1.1.0/30). On Router 1:

```
Router1(config)# interface Tunnel1
Router1(config-if)# ip address 10.1.1.1 255.255.255.252
Router1(config-if)# tunnel source Loopback0
Router1(config-if)# tunnel destination 200.0.0.2
Router1(config-if)# tunnel mode gre ip
```

Configure also the overlay network routing by activating a second OSPFv2 process (process 2) within the overlay network (tunnel network and private networks), on Router 1:

```
Router1(config-if)# interface Tunnel1
Router1(config-if)# ip ospf 2 area 0
Router1(config-if)# interface FastEthernet0/1
Router1(config-if)# ip ospf 2 area 0
```

Perform the equivalent configuration on Router2.

Verify the IPv4 routing tables. Analyze the exchanged OSPFv2 packets exchanged over the overlay network with dual IP headers (networks 10.1.1.0/30). From PC1 ping PC2 while capturing packets on links (R1-RA and RA-R2).

>> Explain how Routers A and B are able to provide connectivity between PC1 and PC2 without knowledge about the private networks.

>> Explain how OSPFv2 process 2 exchange network topology/paths.

>> Explain the importance of multiple OSPF processes in terms of flexibility and redundancy.

3.3. Simulate a network failure by shutting down the interface F0/0 in Router1.

```
Router1(config)# interface f0/0
Router1(config-if)# shutdown
```

Verify the IPv4 routing tables and the status of the tunnel. Retest the connectivity between PC1 and PC2.

>> What do you conclude about the status of the tunnel.

>> Explain the advantage of using loopback interfaces as tunnel end-points versus using physical interfaces as tunnel end-points.