DEPARTAMENTO DE ELETRÓNICA, TELECOMUNICAÇÕES E INFORMÁTICA

LICENCIATURA EM ENGENHARIA DE COMPUTADORES E INFORMÁTICA

ANO 2024/2025

REDES DE COMUNICAÇÕES II

LABORATORY GUIDE NO. 3:

IP TUNNELLING & OVERLAY NETWORKS

In this Laboratory Guide:

- <u>all routers</u> should use the IOS image 15.1(4) of routers 7200 (provided in the elearning page of RC II) and with two network adapters:
  - C7200-IO-2FE in slot 0, providing 2 FastEthernet routing interfaces: `f0/0` and `f0/1`
  - PA-2FE-TX in slot 1, providing 2 FastEthernet routing interfaces: `f1/0` and `f1/1`
- <u>all switches</u> should use the basic Ethernet Switch available in GNS3

1. Initial IPv4 network setup

Create a GNS3 template with all equipment and links of the following network and run the template.

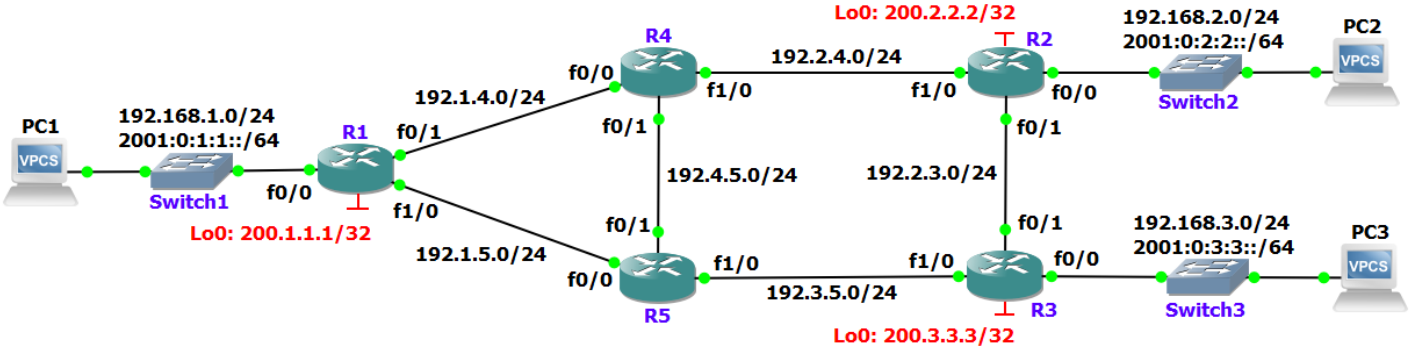
On each interface of each router (including the Loopback interfaces indicated in red in the figure):

- configure an IPv4 address following the IP network address in the figure (in the physical interfaces, use the number of the router name as the host part of the address).
- activate the OSPFv2 protocol in OSPF Process No. 1 and in the backbone area (i.e., area 0).

On each PC:

- configure an IPv4 address following the IP network address in the figure (with the host part of the addresses equal to 100) and the IP address of its default gateway.

By now, ignore IPv6 addresses specified in the figure.



Configuration of IP addresses and activation of OSPFv2 (Process No. 1 and area 0) in router R1:

```
R1# configure terminal
R1(config)# interface f0/0
R1(config-if)# ip address 192.168.1.1 255.255.255.0
R1(config-if)# no shutdown
R1(config-if)# ip ospf 1 area 0

R1(config-if)# interface f0/1
R1(config-if)# ip address 192.1.4.1 255.255.255.0
R1(config-if)# no shutdown
R1(config-if)# ip ospf 1 area 0

R1(config-if)# interface f1/0
R1(config-if)# ip address 192.1.5.1 255.255.255.0
R1(config-if)# no shutdown
R1(config-if)# ip ospf 1 area 0

R1(config-if)# interface loopback 0
R1(config-if)# ip address 200.1.1.1 255.255.255.255
R1(config-if)# no shutdown
R1(config-if)# ip ospf 1 area 0

R1(config-if)# end
R1# write
```

Configuration of IP address and default gateway in PC1:

```
PC1> ip 192.168.1.100/24 192.168.1.1
PC1> save
```

Check the resulting configuration:

```
PC1> show ip
```

1.a. Check the IPv4 routing table of each router. Verify that the routing tables include all existing IPv4 networks and all IPv4 addresses of the Loopback interfaces (if not, there are configuration errors that must be identified and corrected).

Check the complete IPv4 routing table in router R1:

```
R1# show ip route
```

Check the IPv4 routing table in router R1 without the IP addresses of the (L)inks:

```
R1# show ip route | exclude L
```

1.b. Check (through ping) that each PC has IPv4 connectivity with all Loopback interfaces and with all other PCs (if not, there are configuration errors that must be identified and corrected).

2. IP tunnelling

Configure a Tunnel 0 (of type IPv4-IPv4) between interface `f0/1` of router R1 and interface `f1/0` of router R2 and assign the IPv4 network 192.1.2.0/24 to the tunnel (configure the IPv4 address 192.1.2.1 to the endpoint in R1 and the IPv4 address 192.1.2.2 to the endpoint in R2).

Configuration of the endpoint of Tunnel 0 on router R1:

```
R1# configure terminal
R1(config)# interface tunnel 0
R1(config-if)# tunnel source 192.1.4.1        (the address of R1-f0/1)
R1(config-if)# tunnel destination 192.2.4.2   (the address of R2-f1/0)
R1(config-if)# tunnel mode ipip
R1(config-if)# ip address 192.1.2.1 255.255.255.0
R1(config-if)# end
R1# write
```

2.a. Analyze the IPv4 routing tables of the routers and justify the new entries in R1 and R2 related to the configured Tunnel 0.

Start two Wireshark captures: one in network 192.1.4.0/24 and the other in network 192.1.5.0/24.

2.b. In router R1, ping the address 192.1.2.2 (the IP address of the other endpoint of Tunnel 0). Justify the observed ICMP packets in the two networks. Then, analyze the different headers of the ICMP packets and explain how tunnels of type IPv4-IPv4 work.

2.c. Shutdown the interface `f1/0` of router R2 (simulating a link failure). Repeat the ping on router R1 to the address 192.1.2.2. Justify the observed ICMP packets in the two networks. Explain why the ping fails.

Activate the interface `f1/0` of router R2 (i.e., run `no shutdown` on the interface). Reconfigure the source address and the destination address of Tunnel 0 in both routers R1 and R2 with the IP addresses of their Loopback interfaces. Again, start two Wireshark captures (one in network 192.1.4.0/24 and the other in network 192.1.5.0/24).

2.d. In router R1, ping the address 192.1.2.2. Justify the observed ICMP packets in the two networks. Then, analyze the different headers of the ICMP packets to understand how the change of the endpoint source and destination addresses are used by the tunnel.

2.e. Shutdown the interface `f1/0` of router R2 (simulating a link failure). Repeat the ping on router R1 to the address 192.1.2.2. Justify the observed ICMP packets in the two networks. Explain why the ping does not fail when the source and destination addresses of Tunnel 0 are configured with the addresses of the Loopback interfaces.

Activate the interface `f1/0` of router R2 (i.e., run `no shutdown` on the interface). Reconfigure in R1 and R2 the mode of Tunnel 0 to work as a GRE IPv4 type of tunnel (run on each endpoint of Tunnel 0 the command `tunnel mode gre ip`). Again, start two Wireshark captures (one in network 192.1.4.0/24 and the other in network 192.1.5.0/24).

2.f. In router R1, ping the address 192.1.2.2. Justify the observed ICMP packets in the two networks. Then, analyze the different headers of the ICMP packets and explain how tunnels of type GRE IPv4 work (in particular, observe that the GRE header identifies the type of protocol that is in its payload).

In router R1, configure a static IP route to the destination network 192.168.3.0/24 through Tunnel 0:

```
R1# configure terminal
R1(config)# ip route 192.168.3.0 255.255.255.0 tunnel 0
R1(config)# end
R1# write
```

Check the IPv4 routing table of router R1 to verify that the previous OSPF entry to 192.168.3.0/24 was replaced by the configured static route (why?). Again, start two Wireshark captures (one in network 192.1.4.0/24 and the other in network 192.1.5.0/24).

2.g. In PC1, ping the address 192.168.2.100 (the IPv4 address of PC2). Justify the observed ICMP packets in the two networks. Explain why none of the ICMP packets is sent through tunnelling.

2.h. In PC1, ping the address 192.168.3.100 (the IPv4 address of PC3). Analyze the observed ICMP packets in the two networks and explain the differences between this experiment and the previous experiment **2.g**. In particular, explain why the ICMP Echo Request packets are sent through tunnelling while the ICMP Echo Reply packets are not.

Eliminate in router R1 the previously configured static route to use the template in the next task:

```
R1# configure terminal
R1(config)# no ip route 192.168.3.0 255.255.255.0 tunnel 0
R1(config)# end
R1# write
```

3. Overlay networks

The first aim is to create an overlay network to support the IPv6 connectivity between the IPv6 networks of Switch1, Switch2 and Switch 3 (as specified in the figure of page 2). The overlay network will be based on Tunnel 0 (already configured between routers R1 and R2) and a new tunnel, Tunnel 1, to be configured between routers R1 and R3.

First, configure a Tunnel 1 of type GRE IPv4 between routers R1 and R3 (use the Loopback IP addresses as source and destination addresses of the tunnel) and assign the IPv4 network 192.1.3.0/24 to the tunnel (configure the address 192.1.3.1 to the endpoint in R1 and the address 192.1.3.3 to the endpoint in R3).

Then, configure the IPv6 networks on all PCs and on the interfaces `f0/0` of routers R1, R2 and R3, as specified in the figure of page 2 (like in the IPv4 case, use 100 as the host part of the IPv6 addresses in the PCs and use the number of the router name as the host part of the IPv6 addresses in the routers) and activate the IPv6 routing in the three routers.

Configuration of the IPv6 address in PC1:

```
PC1> ip 2001:0:1:1::100/64
PC1> save
```

Check the resulting configuration:

```
PC1> show ipv6
```

IPv6 addressing configuration and routing activation in router R1:

```
R1# configure terminal
R1(config)# ipv6 unicast-routing
R1(config)# interface f0/0
R1(config-if)# ipv6 address 2001:0:1:1::1/64
R1(config-if)# end
R1# write
```

By now, the three IPv6 networks are isolated. To use the overlay network to support the IPv6 connectivity between these IPv6 networks, each tunnel of the overlay network must be assigned with a new IPv6 network address:

- Assign 2001:12::/64 to Tunnel 0 (configure the IPv6 address 2001:12::1 to the endpoint in R1 and the IPv6 address 2001:12::2 to the endpoint in R2).
- Assign 2001:13::/64 to Tunnel 1 (configure the IPv6 address 2001:13::1 to the endpoint in R1 and the IPv6 address 2001:13::3 to the endpoint in R3).

Configuration of the IPv6 addresses of both tunnel endpoints on router R1:

```
R1# configure terminal
R1(config)# interface tunnel 0
R1(config-if)# ipv6 address 2001:12::1/64
R1(config-if)# interface tunnel 1
R1(config-if)# ipv6 address 2001:13::1/64
R1(config-if)# end
R1# write
```

3.a. Check the IPv6 routing table of routers R1, R2 and R3. Verify that the routing tables include all directly connected IPv6 networks (if not, there are configuration errors that must be identified and corrected).

Check the complete IPv6 routing table in router R1:

```
R1# show ipv6 route
```

Check the IPv6 routing table in router R1 without the IPv6 addresses of the (L)inks:

```
R1# show ipv6 route | exclude L
```

Start two Wireshark captures (one in network 192.1.4.0/24 and the other in network 192.1.5.0/24).

3.b. In router R1, first ping the address 192.1.2.2 (the IPv4 address of the other endpoint of Tunnel 0) and then ping the address 2001:12::2 (the IPv6 address of the other endpoint of Tunnel 0). Justify the observed ICMP packets in the two networks. Then, analyze the different headers of the ICMP packets of both pings and explain the usefulness of tunnels of type GRE IPv4.

3.c. Repeat the previous experiment **3.b** but now pinging the IPv4 and IPv6 addresses of the other endpoint of Tunnel 1. Justify the differences between this experiment and the previous experiment.

To reach full IPv6 connectivity, activate the OSPFv3 routing protocol in all IPv6 interfaces of routers R1, R2 and R3. Consider all interfaces in the OSPF Process No. 1 and in the backbone area 0.

Activation of OSPFv3 in all IPv6 interfaces of router R1 with OSPF Process No. 1 and area 0:

```
R1# configure terminal
R1(config)# interface f0/0
R1(config-if)# ipv6 ospf 1 area 0
```

```
R1(config-if)# interface tunnel 0
R1(config-if)# ipv6 ospf 1 area 0
R1(config-if)# interface tunnel 1
R1(config-if)# ipv6 ospf 1 area 0
R1(config-if)# end
R1# write
```

3.d. Check the IPv6 routing table of routers R1, R2 and R3 and justify their entries. Verify that the routing tables include all existing IPv6 networks (if not, there are configuration errors that must be identified and corrected).

3.e. Check (through ping) that there is IPv6 connectivity between all PCs (if not, there are configuration errors that must be identified and corrected).

Start two Wireshark captures (one in network 192.1.4.0/24 and the other in network 192.1.5.0/24).

3.f. In PC2, ping the address 2001:0:1:1::100 (the IPv6 address of PC1). Justify the observed ICMP packets in the two networks and explain the different headers of each observed packet.

3.g. Repeat the previous experiment **3.f** but now pinging in PC2 the IPv6 address of PC3. Justify the differences between this experiment and the previous experiment.

The second aim is to configure the overlay network (composed by the configured Tunnels 0 and 1) **to support also the connectivity between the IPv4 private network addresses of Switch1, Switch2 and Switch 3** (by default, such IPv4 networks are not routed in public IPv4 networks).

First, exclude the IPv4 private network addresses from the OSPFv2 Process No. 1 in routers R1, R2 and R3 (run command `no ip ospf 1 area 0` on the interfaces `f0/0` of routers R1, R2 and R3).

3.h. Check the IPv4 routing table of each router. Verify that the IPv4 private network addresses are only known in the directly connected routers (if not, there are configuration errors that must be identified and corrected).

To reach full connectivity between the IPv4 private networks, activate a new process of the OSPFv2 routing protocol in interfaces `f0/0` of routers R1, R2 and R3 and in all tunnel endpoints (consider all interfaces in the OSPF Process No. 2 and in the backbone area 0).

Activation of OSPFv2 on router R1 with OSPF Process No. 2 and area 0:

```
R1# configure terminal
R1(config)# interface f0/0
R1(config-if)# ip ospf 2 area 0
R1(config-if)# interface tunnel 0
R1(config-if)# ip ospf 2 area 0
R1(config-if)# interface tunnel 1
R1(config-if)# ip ospf 2 area 0
R1(config-if)# end
R1# write
```

3.i. Check (through ping) that there is IPv4 connectivity between all PCs (if not, there are configuration errors that must be identified and corrected).

3.j. Analyze and justify the IPv4 routing tables of routers R1, R2 and R3.

Start two Wireshark captures (one in network 192.1.4.0/24 and the other in network 192.1.5.0/24).

3.k. In PC2, ping the address 192.168.1.100 (the IPv4 address of PC1). Justify the observed ICMP packets in the two networks and explain the different headers of each observed packet.

3.l. Repeat the previous experiment **3.k** but now pinging in PC2 the IPv4 address of PC3. Justify the differences between this experiment and the previous experiment.