FUNDAMENTOS DE REDES

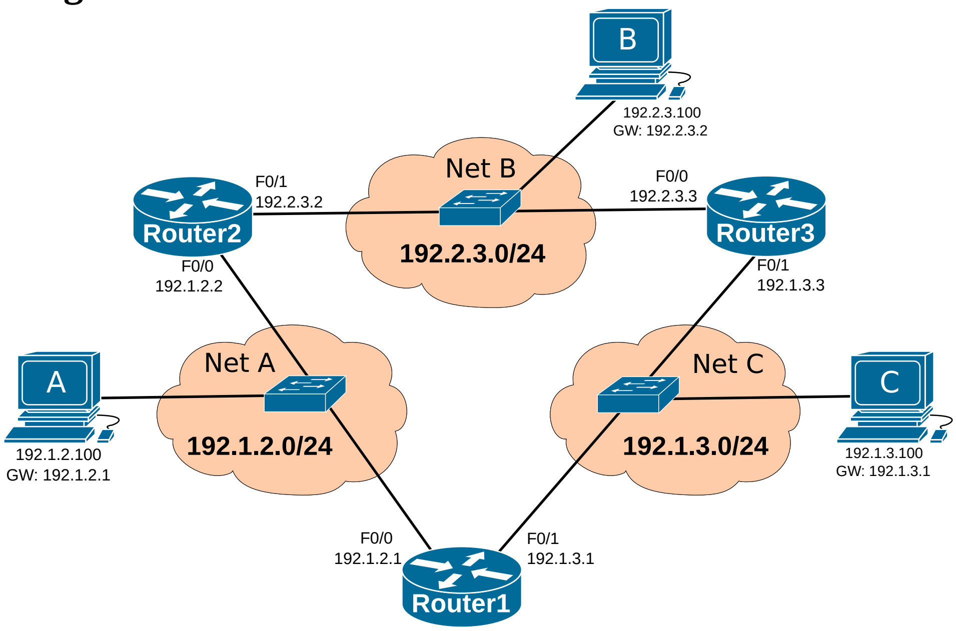
Objectives

- Configuration and analysis of IPv4 and IPv6 static routing.
- Configuration and analysis of RIPv2 and RIPng protocols.
- Configuration and analysis of OSPFv2 and OSPFv3 protocols.

Duration

- 3 weeks

IPv4 Static Routing



6.Assemble the above depicted network **using GNS3** and configure all IPv4 addresses. Verify the interfaces' configurations, routing tables, and IPv4 ARP tables:

```
Router1# show ip interface brief
Router1# show ip route
Router1# show ip arp
```

Start a packet capture in all networks (PC-SW connection), and execute multiple ping commands to test the connectivity between the different equipment's interfaces.
>> Explain the reasons to have connectivity between some interfaces and not others.
>> Explain the reasons why some *ICMP Requests* reach the destination network but the ping is not successful.
>> Explain why *ICMP Destination Unreachable* packets are sent sometimes by the routers.

7. Configure at Router1 a static route to network B via Router2:

```
Router1(config)# ip route 192.2.3.0 255.255.255.0 192.1.2.2
```

Execute a ping command on PC A for the 3 IPv4 addresses of network B.
>> Justify the resulting routing tables on all routers.
>> Explain the reason why some IPv4 addresses can be reached and others not.
>> Explain why an *ICMP Redirect* packet is sometimes sent by Router 1.

8. Configure one static route on Router 2 and another one on Router 3 in order to obtain full connectivity. Register and justify the resulting routing tables of all routers. With the *ping* command, verify that all PCs have connectivity with all IPv4 addresses.

9. Disable (with command `shutdown`) the interface of Router 2 with network B (simulating, in this way, an interface failure). Register and justify the routing table of Router 1. Execute a ping command from Router 1 to PC B.
>> Explain the main disadvantage of using static routes.

10. **Restore the disconnected interface and remove all static routes** configured in the previous experiments (`no ip route *`).

IPv4 Dynamic Routing with RIPv2

11. Configure the RIPv2 protocol at all routers. For Router 1:

```
Router1(config)# router rip
Router1(config-router)# version 2
Router1(config-router)# network 192.1.2.0
Router1(config-router)# network 192.1.3.0
```

Repeat similar configurations on Routers 2 and 3. Verify that the network has full connectivity.
>> Verify and justify the routing tables obtained at all routers.

12. **Start a packet capture in all networks (PC-SW connection)** and captured at least 6 RIP packets in each network. Analyze the captured packets
>> Determine the type of RIP packets sent by each router and its periodicity.
>> Determine the IPv4 destination address, transport protocol and port numbers used by RIPv2.
>> Determine if *split-horizon* is being used or not (default setting).
>> Explain how RIPv2 works.

13. Invert the *split-horizon* configuration at the routers' interfaces that are connected to network A.
To turn *split-horizon* on an interface:
```
Router1(config)# interface F0/0
Router1(config-if)# ip split-horizon
```
To turn *split-horizon* off on an interface:
```
Router1(config)# interface F0/0
Router1(config-if)# no ip split-horizon
```
Capture again at least 6 packets. Analyze the captured packets and compare their content with the packets captured in the previous experiment.
>> Justify the content of RIPv1 packets when *split-horizon* is used and when it is not used.
Restore the *split-horizon* default configuration at the router interfaces connected to network A.

14. **Start a packet capture in all networks (one of the Router-SW connections).** Execute ping commands from Router 1 to all IPv4 addresses of network B. From the packet captures, infer the routing paths followed by the ICMP packets after each ping command.
>> Explain why the packets (from different ping commands) followed different paths.

15. **Start a packet capture in network A (Router 1-SW connection).** Disable (with command `shutdown`) the interface of Router 1 with network C
>> Register and justify the routing tables obtained at all routers.
>> By analyzing the captured packets, explain how RIP protocol propagated the information after the failure.

16. **Start a packet capture in network A (Router 1-SW connection).** Re-enable (with command `no shutdown`) the interface of Router 1 with network C
>> Register and justify the routing tables obtained at all routers.
>> By analyzing the captured packets, explain how RIP protocol propagated the information after the failure recovery.

17. Configure at Router1 a static route to network B via Router2:
```
Router1(config)# ip route 192.2.3.0 255.255.255.0 192.1.2.2
```
>> Register and justify the routing tables obtained at all routers.
>> Explain why Router 1 does not have any routes to network B learned with RIP.

18. Re-configure at Router1 the static route to network B via Router2 to have an administrate distance of 200:
```
Router1(config)# no ip route 192.2.3.0 255.255.255.0 192.1.2.2
Router1(config)# ip route 192.2.3.0 255.255.255.0 192.1.2.2 200
```
>> Register and justify the routing tables obtained at all routers.
>> Explain how the administrative distance works.

19. **Start a packet capture in all networks (PC-SW connection)** and captured at least 6 RIP packets in each network. Assuming that Router 1 provides Internet connectivity, configure it to announce a default route via RIPv2:
```
Router1(config)# router rip
Router1(config-router)# default-information originate
```
>> Register and justify the routing tables obtained at all routers.
>> Explain how an IPv4 packet destined to a network not known by the routers is routed.
>> Analyze the captured RIPv2 packets and explain how the default route is announced.

20 (optional). Activate RIPv2 MD5 Authentication mode (with password *labcom*) in both interfaces of Router 1:

```
Router1(config)# key chain key_chain
Router1(config-keychain)# key 1
Router1(config-keychain-key)# key-string labcom
Router1(config)# interface range F0/0 - 1
Router1(config-if-range)# ip rip authentication key-chain key_chain
Router1(config-if-range)# ip rip authentication mode md5
```

>> Register and justify the routing tables obtained at all routers.

21 (optional). **Start a packet capture in all networks (PC-SW connection)** and captured at least 6 RIP packets in networks A and C. Analyze the captured packets

>> Analyze the RIPv2 packets sent by Router 1, namely the authentication data field.

22 (optional). Activate RIPv2 MD5 Authentication mode (with password *labcom*) in the other Routers' interfaces:

>> Register and justify the routing tables obtained at all routers.

>> Justify the importance of using authentication.

23. **Disable RIP protocol in all routers:**

```
      Router1(config)# no router rip
--
      Router2(config)# no router rip
--
      Router3(config)# no router rip
```

IPv4 Dynamic Routing with OSPFv2

24. Activate OSPFv2 protocols (process number 1, single area) in all routers interfaces:

```
Router1(config)# interface F0/0
Router1(config-if)# ip ospf 1 area 0
```

Repeat similar configurations on other interfaces and Routers 2 and 3.

>> Analyze and justify the routing table of all routers.

>> Retest the connectivity between the equipment.

25. Run the following commands in all routers:

```
Router # show ip ospf
Router# show ip ospf interface
Router# show ip ospf interface brief
Router# show ip ospf neighbor
Router# show ip ospf neighbor detail
```

>> Verify each router OSPF router IDs, identify the DR and BDR of each LAN, and if the cost values assigned by default to each interface agree with the costs of the routing table paths.

>> Explain the choice of DR/BDR.

Note: DR → Designated Router and BDR → Backup Designated Router.

26. Manually define OSPF router IDs:

```
 Router1(config)# router ospf 1
 Router1(config-router)# router-id 1.1.1.1
--
 Router2(config)# router ospf 1
 Router2(config-router)# router-id 2.2.2.2
--
 Router3(config)# router ospf 1
 Router3(config-router)# router-id 3.3.3.3
```

Save all configurations, stop the routers, and restart all routers. Re-verify the OSPF router IDs and re-identify the DR and BDR of each LAN.

>> Explain the differences (if any) on the choice of DR/BDR.

27. Start a capture on link Router1-Router3. In order to verify the bootstrap process of OSPF, reset the OSPFv2 process in Router1 with the command:

```
Router1# clear ip ospf 1 process
```

Analyze the exchanged OSPFv2 packets (with emphasis on the LS types) and explain their contents.
Wait for 1 minute and shutdown the FastEthernet0/1 interface on Router1 (SW1 network). Wait for 1 minute and reactivate (`no shutdown`) the interface on Router1.
>> Analyze the exchanged OSPFv2 packets.

28. Analyze the *Router Link States* and *Network Link States* database information. To view the OSPF databases use the commands:

```
show ip ospf database                      !for a database summary
show ip ospf database router               !for the  Router Link States
show ip ospf database network              !for Network Link States
```

>> Identify all network topology elements within OSPF database.

29. Change the OSPF cost of Router2's interface F0/0 to 50:

```
Router2(config)# interface FastEthernet 0/0
Router2(config-if)# ip ospf cost 50
```

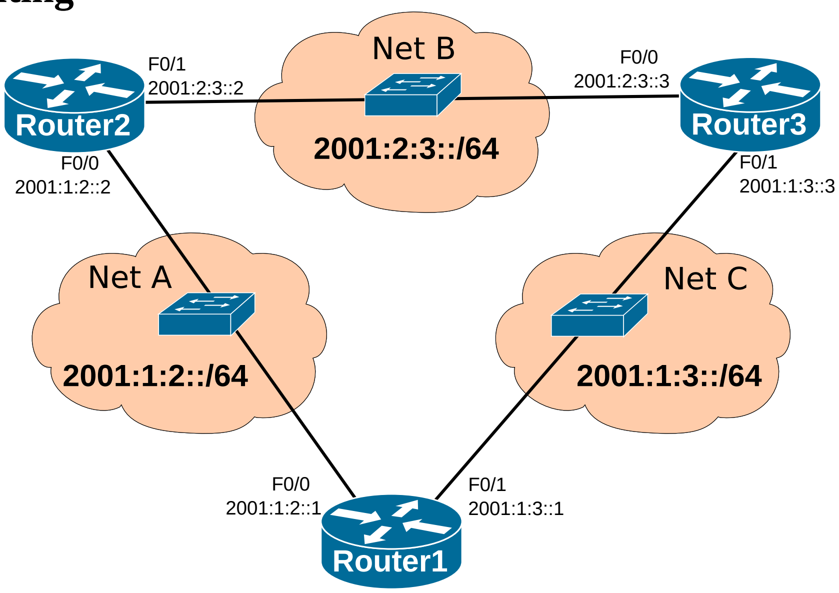
>> Analyze and justify the changes in the routing tables of all routers.
>> Analyze the captured OSPF packets during the network changes.

30. Configure Router 1 to announce a default IPv4 route:

```
Router1(config)# router ospf 1
Router1(config-router)# default-information originate always
```

>> Re-verify the routing tables and identify the default route being announced by Router1.

IPv6 Static Routing



31. Configure all IPv6 addresses and activate IPv6 routing with the command:

```
Router1(config)# ipv6 unicast-routing
```

Verify the interfaces' IPv6 addresses and status, routing tables, IPv6 neighbors and running IPv6 protocols:

```
Router1# show ipv6 interface brief
Router1# show ipv6 route
Router1# show ipv6 neighbors
Router1# show ipv6 protocols
```

Start a packet capture in all networks (Router-SW connection), and execute multiple ping commands to test the connectivity between the different equipment's interfaces.
>> Explain the reasons to have connectivity between some interfaces and not others.

32. Configure all necessary static routes in order to achieve full IPv6 connectivity:

```
Router(config)# ipv6 route <ipv6-net> <ipv6_next_hop>
```

>> Reverify the routing tables and retest the connectivity between the equipments.

IPv6 Dynamic Routing with RIPng

33. <u>Remove all static routes that were configured in the previous experiment.</u>
Activate the RIPng protocol (process named proc1) in all routers interfaces:

```
Router1(config)# interface range F0/0 - 1
Router1(config-if-range)# ipv6 rip proc1 enable
```

Repeat similar configurations on Routers 2 and 3.
Verify also the RIPng information in routers:

```
show ipv6 rip proc1
show ipv6 rip proc1 database
show ipv6 rip proc1 next-hops
```

>> Re-verify the routing tables and retest the connectivity between the equipments.

34. **<u>Start a packet capture in all networks (Router-SW connection)</u>** and captured at least 6 RIPng packets in each network. Analyze the captured packets
>> Determine the type of RIPng packets sent by each router and its periodicity.
>> Determine the IPv6 source and destination addresses, transport protocol and port numbers used by RIPng.
>> Explain how RIPng works.

35. **<u>Start a packet capture in all networks (Router-SW connection)</u>** and disable the *split-horizon* configuration in Router 1.
To turn *split-horizon* off:

```
Router1(config)# ipv6 router rip proc1
Router1(config-rtr)# no split-horizon
```

Capture again at least 6 packets. Analyze the captured packets and compare their content with the packets captured in the previous experiment.
>> Justify the content of RIPng packets when *split-horizon* is used and when it is not used.
Restore the *split-horizon* default configuration in Router 1.

36. **<u>Start a packet capture in all networks (Router-SW connection)</u>** and captured at least 6 RIPng packets in each network. Assuming that Router 1 provides Internet connectivity, configure it to announce a default route via RIPng:

```
Router1(config)# interface range F0/0 - 1
Router1(config-if-range)# ipv6 rip proc1 default-information originate
```

>> Register and justify the routing tables obtained at all routers.
>> Explain how an IPv6 packet destined to a network not known by the routers is routed.
>> Analyze the captured RIPng packets and explain how the default route is announced

IPv6 Dynamic Routing with OSPFv3

37. Activate OSPFv3 protocols (process number 1, single area) in all routers interfaces:

```
Router1(config)# interface F0/0
Router1(config-if)# ipv6 ospf 1 area 0
```

Repeat similar configurations on other interfaces and Routers 2 and 3.
Analyze also the OSPFv3 information in routers:

```
show ipv6 ospf 1 database
show ipv6 ospf 1 database network
show ipv6 ospf 1 database router
show ipv6 ospf 1 database prefix              !New OSPFv3 database
```

>> Re-verify the routing tables and retest the connectivity between the equipment.

37. Start a capture on link Router1-Router3. In order to verify the bootstrap process of OSPFv3, reset the OSPFv3 process in Router1 with the command:

```
Router1# clear ipv6 ospf 1 process
```

>> Analyze the exchanged OSPFv3 packets.

38. Configure Router 1 to announce a default IPv6 route:

```
Router1(config)# ipv6 router ospf 1
Router1(config-router)# default-information originate always
```

>> Re-verify the routing tables and identify the default route being announced by Router1.