Lab 04: Implementing Static and Dynamic Routing (RIP & OSPF)

Duration: 1 hours 30 minutes | In-Lab Assessment
Tools: Cisco Packet Tracer

Objectives:

- Define and describe the concept of routing
- Describe the concept of static routing and dynamic routing
- Configure static and dynamic routing in a given network topology

Lab Description:

Part A: Static Routing

Static routing requires the administrator to manually configure and update all routes in a router's table—changes in the network (like a link failure) must be addressed by hand, unlike dynamic routing, which adapts automatically.

Because it doesn't exchange update packets, static routing uses far fewer CPU cycles and less bandwidth, making it a lightweight backup solution in large networks rather than the primary method.

However, it demands complete knowledge of the network's topology and addresses, so it's impractical to use alone at scale. To handle packets whose destinations lack a specific entry, routers employ a default route, which directs such traffic out a predefined interface.

When working with CISCO devices, specifically for this lab, you will encounter two types of static default routes. They are **directly connected** static default route and **next-hop** static default route. The general format of the static routes is as follows:

```
ip route destination_network_prefix destination_prefix_mask (next-hop_address
| interface) [distance_metric]
```

To configure the directly connected static default routes, you will specify the **interface**. For the next-hop static default routes, you will specify the **next_hop address**. One special use case of the above command is configuring a **primary static default route** where both the `destination_network_prefix` and `Destination_prefix_mask` are 0.0.0.0. The IPv4 command format for specifying the primary static default route is given below:

```
ip route 0.0.0.0 0.0.0.0 (next-hop_address | interface) [distance_metric]
```

It means that **"packet from any IP address with any subnet mask gets sent to the specified next-hop address or interface."**

Another concept when configuring static routing is the **floating static route**. A floating route is nothing but the route that is used to forward a packet to a certain destination when main route is unavailable. The floating routes are defined by providing a higher **distance_metric** to a certain route. The default distance_metric, when it is not manually specified, is 1. The floating static routes are given higher numbers than 1. Routers always take the route with lower distance_metric when multiple routes to the same destination are available. That's why this floating static route will only be used when the main route (distance_metric is 1) is down or unavailable.

Part B: Dynamic Routing

Dynamic routing lets routers select routes based on the real-time network condition. There will always be packets traveling around the networks to keep the routers up-to-date about the present network condition. Then, the routers will select an optimal route to a given destination based on some set of metrics. There are different types of dynamic routing protocols following different algorithms. The two most common ones are **Routing Information Protocol (RIP)** following distance-vector algorithm and **Open Shortest Path First (OSPF)** protocol following link state routing algorithm.

Routing Information Protocol (RIP)

RIP is somewhat obsolete due to its limitations and the advent of more modern and sophisticated protocols like OSPF, EIGRP, etc. Still, as this was one of the pioneering dynamic routing protocols that dominated the networking world for quite some time, you must understand this. This will help you understand the improvements made in newer protocols. There are two versions of RIP, namely RIPv1 and RIPv2. RIPv2 is the dominant one and is used in almost all cases where RIP is used. So, we'll focus on RIPv2 only for our lab. A major problem of RIPv1 was that it used to broadcast routing table updates every 30 seconds. You can imagine the flood of packets every 30 seconds that would take place where millions of networks are now interconnected, even if they are configured to send updates at random times. RIPv2 was designed to overcome this issue along with other improvements. You can learn more about these two versions along with RIPng (for IPv6) here.

Open Shortest Path First (OSPF) Routing Protocol

OSPF uses a Link-State algorithm: routers flood only their link states (not full tables) to neighbors, then each runs Dijkstra's Shortest-Path First to build routes. This makes OSPF more CPU-intensive than distance-vector protocols, but it converges faster and balances load better than RIP.

Key OSPF Concepts

1. **Area:** A logical grouping of routers sharing topology info. All non-backbone areas (any ID != 0) connect through the backbone (Area 0).
2. **Area Border Router (ABR):** Connects two areas and exchanges summaries between them.
3. **Autonomous System (AS):** A set of OSPF areas under one administrative domain.
4. **Designated Router (DR)/Backup DR:** On multi-access links (e.g. Ethernet), the DR/BDR reduce LSA traffic by acting as a hub for broadcasts.

5. **Router ID:** A unique 32-bit identifier (manually set or chosen as highest loopback/active IP). Requires process reset (clear ip ospf process).
6. **Cost:** Inverse of bandwidth: cost = reference-bandwidth / interface-bandwidth (default ref = 100 Mbps). Adjust with auto-cost reference-bandwidth and bandwidth commands.
7. **Wildcard Mask:** Used in network <IP> <wildcard> area <ID> to select interfaces—0 bits must match, 1 bits ignore (e.g. 0.0.0.255 for /24).
8. **Process ID:** Local identifier for an OSPF instance (router ospf <process_id>); multiple instances can run on one router.

Neighbor Maintenance

1. **Hello and Dead Intervals:** Hello packets sent every X seconds (default 10); if no hello is heard in 4 × Hello, the neighbor is declared down.
2. **Passive Interfaces:** Advertise networks but do not send hellos (common on LANs).

Part C: Route Redistribution

Redistributing routing information refers to the process of sharing routing information between different routing protocols within a network. The messages used to advertise or update routing information are protocol-specific; hence, information from one protocol cannot be directly used by another to update its routing information.

For instance, the router at the boundary of the OSPF and RIP networks can be set up to redistribute routes as follows:

Redistributing OSPF into RIP: When the OSPF router receives a route, it advertises it to the RIP domain, allowing branch offices using RIP to learn about routes to the internal OSPF network.

Redistributing RIP into OSPF: Conversely, the router can also redistribute routes learned from RIP into the OSPF network, ensuring that the internal OSPF network can reach the branch offices.

Part D: Configure static routing

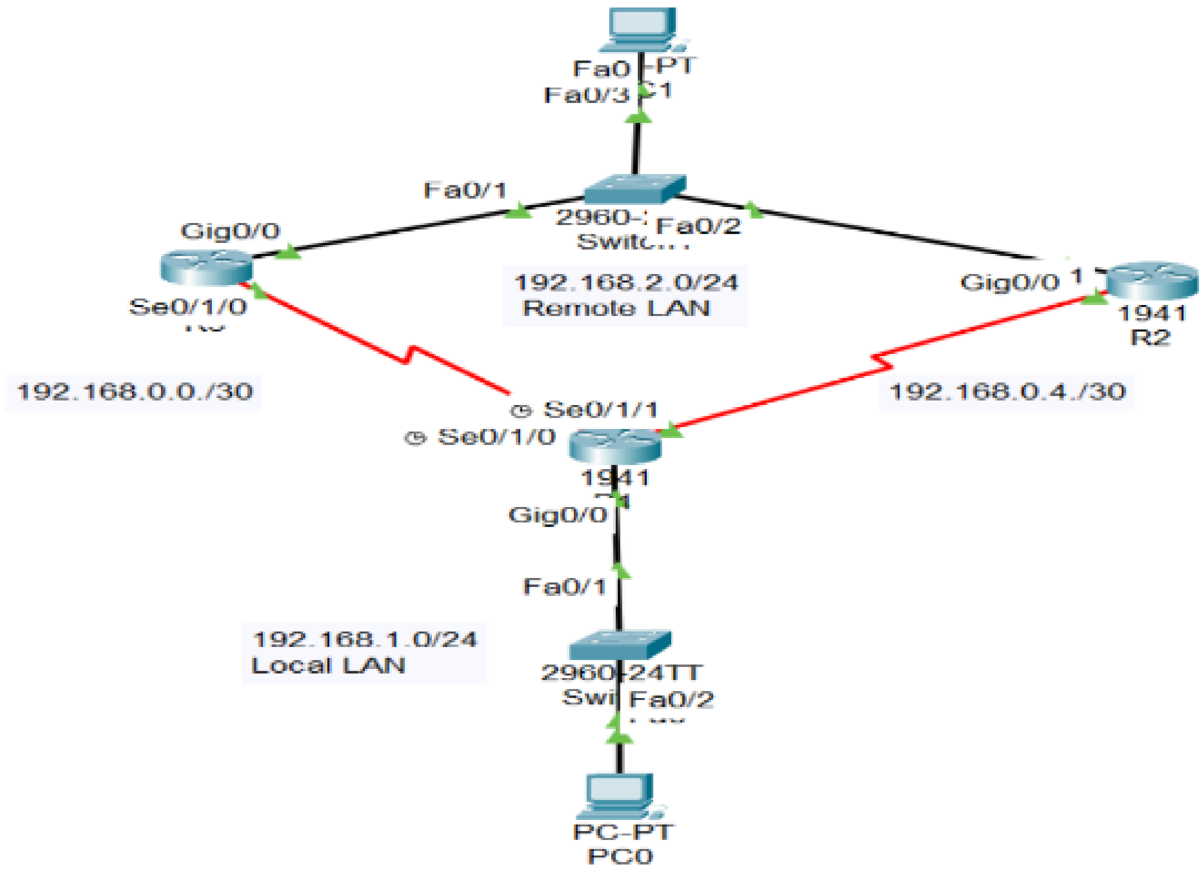

Figure 4: A sample network topology for the configuration of static routing.

a. Configure R1 Interfaces

```
R1(config)#int g0/0
R1(config-if)#ip add 192.168.1.1 255.255.255.0
R1(config-if)#desc connection-to-PC0
R1(config-if)#no shutdown
R1(config-if)#exit
R1(config)#int s0/1/0
R1(config-if)#ip add 192.168.0.2 255.255.255.252
R1(config-if)#desc connection-to-R3
R1(config-if)#clock rate 64000
R1(config-if)#no shutdown
R1(config-if)#exit
R1(config)#int s0/1/1
R1(config-if)#ip add 192.168.0.6 255.255.255.252
R1(config-if)#desc connection-to-R2
R1(config-if)#clock rate 64000
R1(config-if)#no shutdown
R1(config-if)#exit
```

b. Configure R2 Interfaces

```
R2(config)#int s0/1/1
R2(config-if)#ip add 192.168.0.5 255.255.255.252
R2(config-if)#desc connection-to-R1
R2(config-if)#no shutdown
R2(config-if)#exit
R2(config)#int g0/0
R2(config-if)#ip add 192.168.2.1 255.255.255.0
R2(config-if)#desc connection-to-RemoteLAN
R2(config-if)#no shutdown
R2(config-if)#exit
```

c. Configure R3 Interfaces

```
R3(config)#int s0/1/0
R3(config-if)#ip add 192.168.0.1 255.255.255.252
R3(config-if)#desc connection-to-R1
R3(config-if)#no shutdown
R3(config-if)#exit
R3(config)#int g0/0
R3(config-if)#ip add 192.168.2.2 255.255.255.0
R3(config-if)#desc connection-to-RemoteLAN
R3(config-if)#no shutdown
R3(config-if)#exit
```

d. Configure PC0

```
IP: 192.168.1.10
Mask: 255.255.255.0
Gateway: 192.168.1.1
```

e. Configure PC1

```
IP: 192.168.2.10
Mask: 255.255.255.0
Gateway: 192.168.2.1
```

f. Configure static routing to Remote LAN in R1

```
R1(config)#ip route 192.168.2.0 255.255.255.0 s0/1/1
```

It's a **directly connected** static default route.

```
R1(config)#ip route 192.168.2.0 255.255.255.0 192.168.0.1 5
```

It's a **next-hop floating** static default route.

g. Configure static routing to Local LAN in R2

```
R2(config)#ip route 192.168.1.0 255.255.255.0 s0/1/1
```

It's a **directly connected** static default route.

```
R2(config)#ip route 192.168.1.0 255.255.255.0 192.168.0.6 5
```

It's a **next-hop floating** static default route.

h. **Configure static routing to Local LAN in R3**

```
R2(config)#ip route 192.168.1.0 255.255.255.0 s0/1/0
```

It's a **directly connected** static default route.

```
R2(config)#ip route 192.168.1.0 255.255.255.0 192.168.0.2 5
```

It's a **next-hop floating** static default route.

i. **Verify**

```
Ping PC1 from PC0
```

Part E: Configure RIP

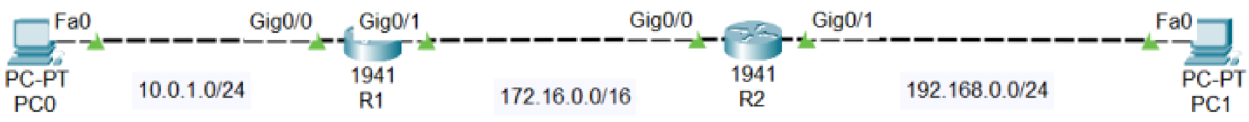

Figure 5: A sample network topology for the configuration of RIP.

a. **Configure R1 Interfaces**

```
R1(config)# int g0/0
R1(config-if)# ip address 10.0.1.1 255.255.255.0
R1(config-if)# no shutdown
R1(config-if)# exit
R1(config)# int g0/1
R1(config-if)# ip address 172.16.0.1 255.255.0.0
R1(config-if)# no shutdown
R1(config-if)# exit
R1# copy running-config startup-config
```

b. **Configure R2 Interfaces**

```
R2(config)# int g0/1
R2(config-if)# ip address 192.168.0.1 255.255.255.0
R2(config-if)# no shutdown
R2(config-if)# exit
R2(config)# int g0/0
R2(config-if)# ip address 172.16.0.2 255.255.0.0
R2(config-if)# no shutdown
R2(config-if)# exit
R2# copy running-config startup-config
```

c. **Configure PC0**

```
IP: 10.0.1.10
Mask: 255.255.255.0
Gateway: 10.0.1.1
```

d. **Configure PC1**

```
IP: 192.168.0.10
Mask: 255.255.255.0
Gateway: 192.168.0.1
```

e. **Configure RIP in R1**

```
R1(config)#router rip
R1(config-router)#version 2
R1(config-router)#network 10.0.0.0
R1(config-router)#network 172.16.0.0
```

f. **Configure RIP in R2**

```
R2(config)#router rip
R2(config-router)#version 2
R2(config-router)#network 192.168.0.0
R2(config-router)#network 172.16.0.0
```

g. **Verify**

```
R1# show ip route rip

Ping PC1 from PC0
```

Part F: Configure OSPF

The OSPF concepts and the detailed implementation of the OSPF routing protocol can be found at this link. You are suggested to read the article and practice the commands from the article for implementation. Below is a simple configuration of OSPF for the given network topology:

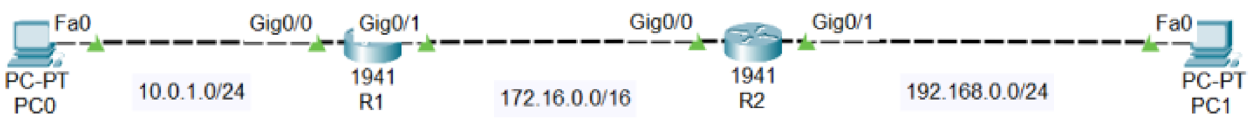

Figure 6: A sample network topology for the configuration of OSPF.

a. **Configure R1 Interfaces**

```
R1(config)# int g0/0
R1(config-if)# ip address 10.0.1.1 255.255.255.0
R1(config-if)# no shutdown
R1(config-if)# exit
```

```
R1(config)# int g0/1
R1(config-if)# ip address 172.16.0.1 255.255.0.0
R1(config-if)# no shutdown
R1(config-if)# exit
R1# copy running-config startup-config
```

b. Configure R2 Interfaces

```
R2(config)# int g0/1
R2(config-if)# ip address 192.168.0.1 255.255.255.0
R2(config-if)# no shutdown
R2(config-if)# exit
R2(config)# int g0/0
R2(config-if)# ip address 172.16.0.2 255.255.0.0
R2(config-if)# no shutdown
R2(config-if)# exit
R2# copy running-config startup-config
```

c. Configure PC0

```
IP: 10.0.1.10
Mask: 255.255.255.0
Gateway: 10.0.1.1
```

d. Configure PC1

```
IP: 192.168.0.10
Mask: 255.255.255.0
Gateway: 192.168.0.1
```

e. Configure OSPF in R1

```
R1(config)# router ospf 1
R1(config-router)# network 10.0.1.0 0.0.0.255 area 0
R1(config-router)# network 172.16.0.0 0.0.255.255 area 0
```

f. Configure OSPF in R2

```
R2(config)# router ospf 1
R2(config-router)# network 192.168.0.0 0.0.0.255 area 0
R2(config-router)# network 172.16.0.0 0.0.255.255 area 0
```

g. Verify

```
R1# show ip ospf neighbor
R2# show ip ospf neighbor

Ping PC1 from PC0
```

Part G: Configure Route Redistribution

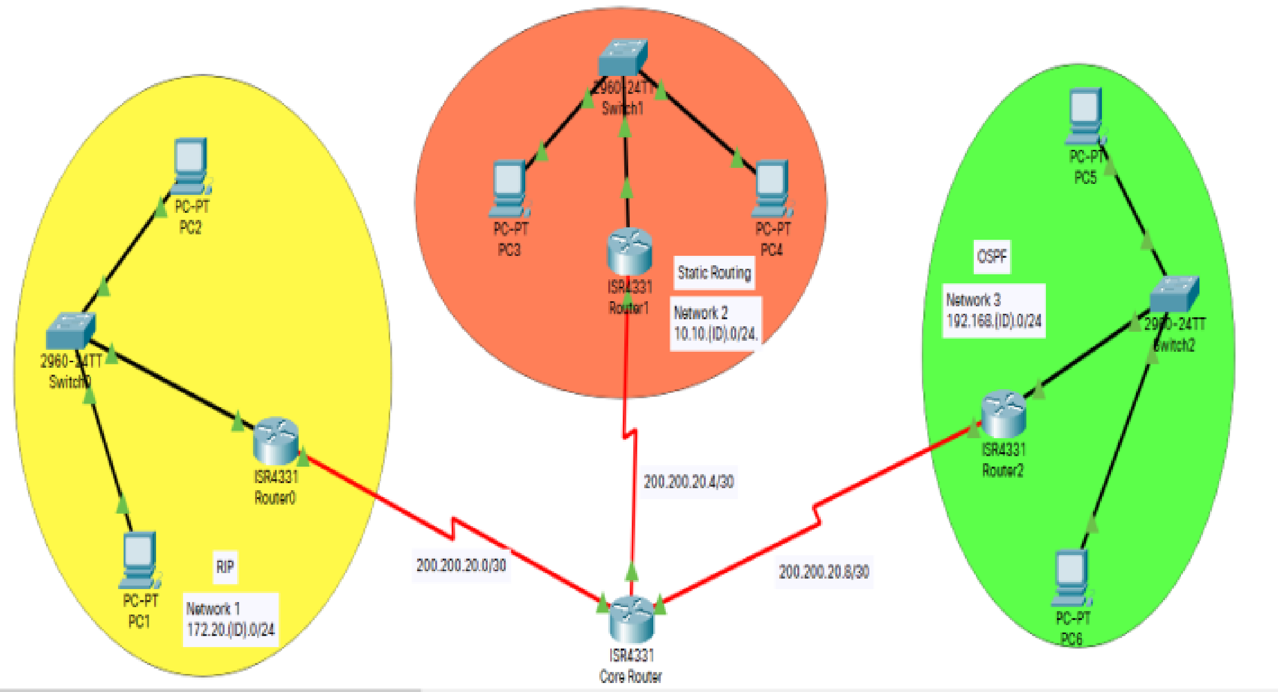

Figure 7: A sample network topology for Route Redistribution.

The core router is running three different routing protocols for the connected subnetworks. Route redistribution is necessary for this network design to enable communication among the subnetworks. It allows routes from one routing protocol to be advertised in another.

a. **Redistribute RIP and Static into OSPF**

```
CoreRouter(config)#router ospf 1
CoreRouter(config-router)#redistribute rip metric 1 subnets
CoreRouter(config-router)#redistribute static metric 1 subnets
```

b. **Redistribute OSPF and Static into RIP**

```
CoreRouter(config)#router rip
CoreRouter(config-router)#redistribute ospf 1 metric 1
CoreRouter(config-router)#redistribute static metric 1
```

c. **Verify**

To verify the route redistribution, use the `ping` messages from **PC1** to **PC5** after successfully assigning the ip addresses to the PCs and routers' interfaces and successfully configuring routing protocols as shown in Figure 7.