Lab 10

Name, Surname	Mnqobi Jeza
Student Number:	230878369
Date:	12 May 2025

Dynamic Routing: OSPF Routing Protocol (Link State Routing Protocol)

The software that will be used is **Cisco Packet Tracer.**

Introduction

OSPF is large and complex and has many attributes. It is also processor-intensive because the Dijkstra algorithm is used to calculate the shortest paths. OSPF is very scalable for large enterprise networks.

1.1. Load lab 3 into packet tracer. It should be the configuration shown in Figure 1 with all IP addresses assigned and all interfaces showing green arrows.

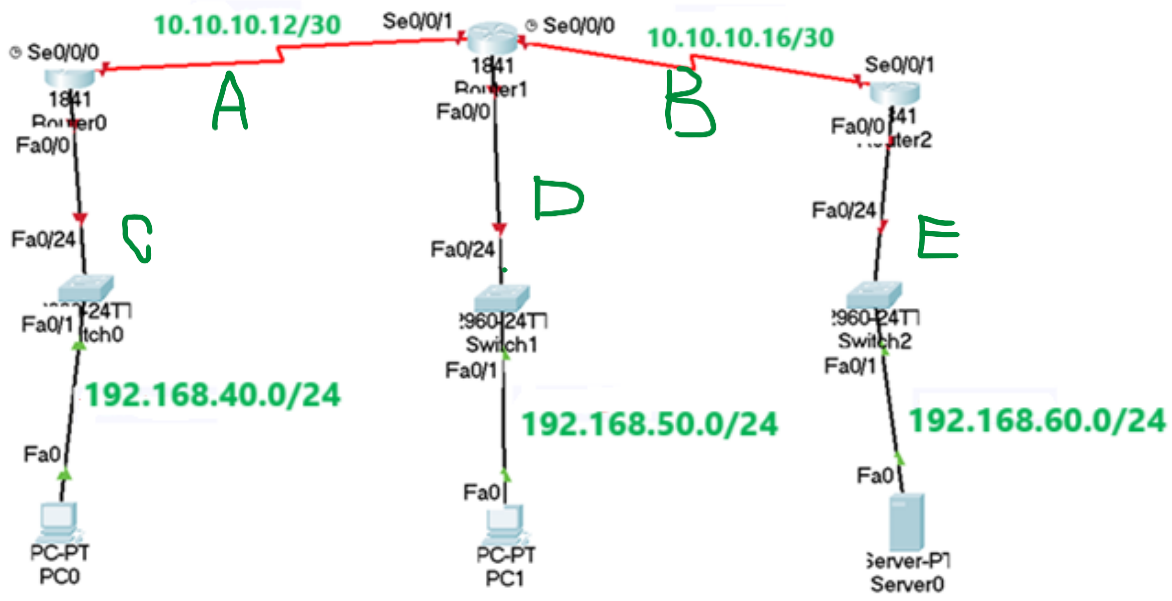


Figure 1: Network Laboratory Topology for IP Assignment

OSPF – Area 0

Process ID number – local to the router (a number used to keep track of the local database for OSPF from 1-65535)

Area: Must be the same on all routers. No more than 50 routers per area.

How to get the wild card mask: Example 1

	255.	255.	255.	255	Constant number
-	255.	255.	255.	0	Subnet Mask
	0.	0.	0.	255	Wildcard Mask

How to get the wild card mask: Example 2

	255.	255.	255.	255	Constant number
-	255.	255.	255.	252	Subnet Mask
	0.	0.	0.	3	Wildcard Mask

Configure OSPF on R1

```
Router0> enable
Router0# configure terminal
Router0(config)# router ospf 1
Router0(config-router)# network networkC 0.0.0.255 area 0
Router0(config-router)# network networkA 0.0.0.3 area 0
Router0(config-router)# exit
Router0(config)# exit
Router0# write memory
```

Now configure R2 and R3 in the same way, using the same area number.

Some thoughts.

- OSPF is a powerful and widely used **link-state routing protocol** designed for medium to large networks. Unlike RIP, which is based on hop count, OSPF makes routing decisions based on **cost**, leading to more efficient path selection.
- The protocol relies on the **Dijkstra algorithm** to compute shortest paths. This is why OSPF can converge quickly, but it is also more **CPU-intensive**, especially on larger networks.
- One of the most important lessons from this lab is the use of **wildcard masks**. You must be able to correctly calculate and apply them—this determines which interfaces participate in OSPF.
- The concept of **OSPF areas** is crucial. In this lab, we worked with a single area (Area 0), which is the **backbone area**. As networks grow, understanding how to design multi-area OSPF becomes vital for scalability and stability.
- Always verify your configuration using commands like:

  show ip ospf neighbor – to check if neighbor relationships are established.
  show ip route – to confirm learned routes via OSPF.
  show ip protocols – to confirm OSPF is running and what networks are advertised.

- Remember, the real value of this lab isn't just about getting green arrows in Packet Tracer. It's about understanding **how routers communicate, exchange routing information, and dynamically adapt** to changes in the network.

Comparing RIPv2 and OSPF

- RIPv2 and OSPF are both dynamic routing protocols, but they differ significantly in how they operate and where they are best used.

- RIPv2 is a distance-vector protocol, meaning it makes routing decisions based on hop count and relies on periodic updates to share routing information. It is simple to configure and good for small networks, but it scales poorly due to its hop limit of 15 and slow convergence.

- OSPF, on the other hand, is a link-state protocol that uses the Dijkstra algorithm to compute the shortest path based on link cost. It is more efficient in large or complex networks due to faster convergence, support for multi-area hierarchies, and better resource usage.

- In terms of routing updates, RIPv2 sends its full routing table every 30 seconds to all neighbors via multicast, which can result in high overhead. OSPF only sends updates when there is a change, and those updates are incremental, making it more bandwidth-efficient.

- RIPv2 is easier for beginners to learn because of its simplicity. However, OSPF introduces students to more advanced networking concepts, such as areas, adjacency states, and LSAs (Link-State Advertisements).

- Practically, learning both protocols allows students to appreciate the evolution of routing protocols—from basic hop-based logic to more intelligent, topology-aware routing.

- Key Takeaway: While RIPv2 is ideal for foundational learning, OSPF equips students with the knowledge needed for real-world, scalable network design. Understanding both helps build a strong foundation in network engineering.

Some debug commands to explore:

- sh start
- sh ip int bri
- sh protocols
- sh ip protocols
- sh ip route
- sh controllers
- debug ip rip

How to submit

2.1. Copy your configuration file for R1 from the CLI and paste it here.

```
R1>show running-config
^
% Invalid input detected at '^' marker.
R1>enable
R1#sh running-config
Building configuration...

Current configuration : 908 bytes
!
version 12.4
no service timestamps log datetime msec
no service timestamps debug datetime msec
no service password-encryption
!
hostname R1
!
!
!
!
!
!
!
!
no ip cef
no ipv6 cef
!
!
!
!
!
!
!
```

```
!
!
!
!
!
spanning-tree mode pvst
!
!
!
!
!
!
interface FastEthernet0/0
ip address 192.168.10.1 255.255.255.0
duplex auto
speed auto
!
interface FastEthernet0/1
no ip address
duplex auto
speed auto
shutdown
!
interface Serial0/0/0
ip address 10.0.0.1 255.255.255.252
clock rate 4000000
!
interface Serial0/0/1
no ip address
clock rate 2000000
shutdown
!
interface Vlan1
no ip address
shutdown
!
router ospf 1
log-adjacency-changes
network 192.168.10.0 0.0.0.255 area 0
network 10.0.0.0 0.0.0.3 area 0
!
router rip
version 2
network 10.0.0.0
network 192.168.10.0
no auto-summary
!
ip classless
!
ip flow-export version 9
!
!
!
```

```
        !
        !
        !
        !
        !
        line con 0
        !
        line aux 0
        !
        line vty 0 4
        login
        !
        !
        !
End

        R2#sh running-config
        Building configuration...

        Current configuration : 930 bytes
        !
        version 12.4
        no service timestamps log datetime msec
        no service timestamps debug datetime msec
        no service password-encryption
        !
        hostname R2
        !
        !
        !
        !
        !
        !
        !
        !
        no ip cef
        no ipv6 cef
        !
        !
        !
        !
        !
        !
        !
        !
        !
        !
        !
        !
        spanning-tree mode pvst
        !
        !
        !
```

```
!
!
!
interface FastEthernet0/0
ip address 192.168.2.1 255.255.255.0
duplex auto
speed auto
!
interface FastEthernet0/1
no ip address
duplex auto
speed auto
shutdown
!
interface Serial0/0/0
ip address 10.0.0.2 255.255.255.252
!
interface Serial0/0/1
ip address 10.0.0.5 255.255.255.252
clock rate 4000000
!
interface Vlan1
no ip address
shutdown
!
router ospf 1
log-adjacency-changes
network 10.0.0.0 0.0.0.3 area 0
network 10.0.0.4 0.0.0.3 area 0
network 192.168.2.0 0.0.0.255 area 0
!
router rip
version 2
network 10.0.0.0
network 192.168.2.0
no auto-summary
!
ip classless
!
ip flow-export version 9
!
!
!
!
!
!
!
!
line con 0
!
line aux 0
!
line vty 0 4
```

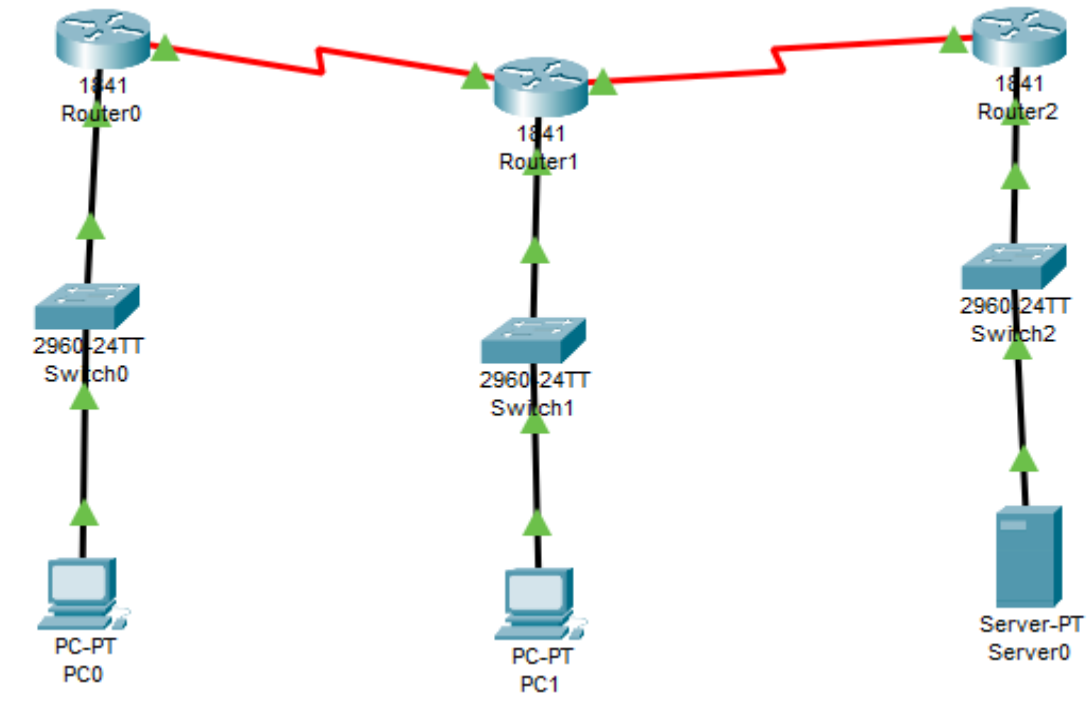
```
        login
        !
        !
        !
End

        R3#sh running-config
        Building configuration...

        Current configuration : 885 bytes
        !
        version 12.4
        no service timestamps log datetime msec
        no service timestamps debug datetime msec
        no service password-encryption
        !
        hostname R3
        !
        !
        !
        !
        !
        !
        !
        !
        no ip cef
        no ipv6 cef
        !
        !
        !
        !
        !
        !
        !
        !
        !
        !
        !
        !
        spanning-tree mode pvst
        !
        !
        !
        !
        !
        !
        interface FastEthernet0/0
        ip address 192.168.1.1 255.255.255.0
        duplex auto
        speed auto
        !
        interface FastEthernet0/1
        no ip address
```

```
 duplex auto
 speed auto
 shutdown
 !
 interface Serial0/0/0
 no ip address
 clock rate 2000000
 shutdown
 !
 interface Serial0/0/1
 ip address 10.0.0.6 255.255.255.252
 !
 interface Vlan1
 no ip address
 shutdown
 !
 router ospf 1
 log-adjacency-changes
 network 10.0.0.4 0.0.0.3 area 0
 network 192.168.1.0 0.0.0.255 area 0
 !
 router rip
 version 2
 network 10.0.0.0
 network 192.168.1.0
 no auto-summary
 !
 ip classless
 !
 ip flow-export version 9
 !
 !
 !
 !
 !
 !
 !
 !
 line con 0
 !
 line aux 0
 !
 line vty 0 4
 login
 !
 !
 !
end
```

2.2. Print the logical topology to the file, then paste the image here. Make sure the topology is big enough so that all elements are legible.



2.3. For R1, show the output of (sh ip route). Paste this output here.

```
R1#sh ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
* - candidate default, U - per-user static route, o - ODR
P - periodic downloaded static route

Gateway of last resort is not set

10.0.0.0/30 is subnetted, 2 subnets
C 10.0.0.0 is directly connected, Serial0/0/0
O 10.0.0.4 [110/128] via 10.0.0.2, 00:05:18, Serial0/0/0
O 192.168.1.0/24 [110/129] via 10.0.0.2, 00:05:18, Serial0/0/0
O 192.168.2.0/24 [110/65] via 10.0.0.2, 00:05:18, Serial0/0/0
C 192.168.10.0/24 is directly connected, FastEthernet0/0
```

2.4. For R2, show the output of (sh ip route). Paste this output here.

```
R2#sh ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
```

```
        i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
        * - candidate default, U - per-user static route, o - ODR
        P - periodic downloaded static route

        Gateway of last resort is not set

        10.0.0.0/30 is subnetted, 2 subnets
        C 10.0.0.0 is directly connected, Serial0/0/0
        C 10.0.0.4 is directly connected, Serial0/0/1
        O 192.168.1.0/24 [110/65] via 10.0.0.6, 00:06:01, Serial0/0/1
        C 192.168.2.0/24 is directly connected, FastEthernet0/0
O 192.168.10.0/24 [110/65] via 10.0.0.1, 00:06:01, Serial0/0/0
```

2.5. For R3, show the output of (sh ip route). Paste this output here.

```
        R3#sh ip route
        Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
        D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
        N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
        E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
        i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
        * - candidate default, U - per-user static route, o - ODR
        P - periodic downloaded static route

        Gateway of last resort is not set

        10.0.0.0/30 is subnetted, 2 subnets
        O 10.0.0.0 [110/128] via 10.0.0.5, 00:06:38, Serial0/0/1
        C 10.0.0.4 is directly connected, Serial0/0/1
        C 192.168.1.0/24 is directly connected, FastEthernet0/0
        O 192.168.2.0/24 [110/65] via 10.0.0.5, 00:06:38, Serial0/0/1
O 192.168.10.0/24 [110/129] via 10.0.0.5, 00:06:28, Serial0/0/1
```

2.6. For R1, show the output of (sh ip int bri). Paste this output here.

```
        R1#sh ip int bri
        Interface IP-Address OK? Method Status Protocol
        FastEthernet0/0 192.168.10.1 YES manual up up
        FastEthernet0/1 unassigned YES unset administratively down down
        Serial0/0/0 10.0.0.1 YES manual up up
        Serial0/0/1 unassigned YES unset administratively down down
Vlan1 unassigned YES unset administratively down down
```

2.7. For R2, show the output of (sh ip int bri). Paste this output here.

```
        R2#sh ip int bri
        Interface IP-Address OK? Method Status Protocol
        FastEthernet0/0 192.168.2.1 YES manual up up
        FastEthernet0/1 unassigned YES unset administratively down down
        Serial0/0/0 10.0.0.2 YES manual up up
```

```
Serial0/0/1 10.0.0.5 YES manual up up
Vlan1 unassigned YES unset administratively down down
```

2.8. For R3, show the output of (sh ip int bri). Paste this output here.

```
R3#sh ip int bri
Interface IP-Address OK? Method Status Protocol
FastEthernet0/0 192.168.1.1 YES manual up up
FastEthernet0/1 unassigned YES unset administratively down down
Serial0/0/0 unassigned YES unset administratively down down
Serial0/0/1 10.0.0.6 YES manual up up
Vlan1 unassigned YES unset administratively down down
```

2.9. For R1, show the output of (sh ip ospf neighbour). Paste this output here.

```
R1#sh ip ospf neighbor

Neighbor ID Pri State Dead Time Address Interface
192.168.2.1 0 FULL/ - 00:00:39 10.0.0.2 Serial0/0/0
```

2.10. For R2, show the output of (sh ip ospf neighbour). Paste this output here.

```
R2#sh ip ospf neighbor

Neighbor ID Pri State Dead Time Address Interface
192.168.10.1 0 FULL/ - 00:00:35 10.0.0.1 Serial0/0/0
192.168.1.1 0 FULL/ - 00:00:35 10.0.0.6 Serial0/0/1
```

END